

iDrive 智能伺服驱动器 用户手册

Version 2.1

北京诺信泰伺服科技有限公司

2010-03

iDrive 智能伺服驱动器

iDrive智能驱动器是诺信泰自主知识产权的运动控制和伺服驱动产品，它秉承了欧美伺服驱动器的特点。iDrive 是一款全数字化分布式智能伺服驱动器，与传统的脉冲伺服驱动产品相比具有诸多优势。iDrive集运动控制、伺服驱动和集成PLC 功能于一个紧凑的单元中，无需外部运动控制器即可实现运动控制功能。iDrive可执行位置、速度、单轴、多轴或独立工作模式的多种运行方式。其分布式的网络结构为您的系统集成带来方便，iDrive的独立运行模式为你特殊的应用带来了更大的节省费用的潜能。我们相信iDrive的诸多优点能为您的成功带来帮助。

定制设计

除了提供标准化的产品，我们还擅长于为大中型规模的应用定制特殊的运动控制和驱动功能。对现有产品的一个简单的修改，就可以满足您对系统的特殊要求，为您节省了大量的时间、精力和费用的同时提高您的生产效率。我们服务于我们的客户，以提供最适合您的产品和最好的服务为己任。

联系方式

北京诺信泰伺服科技有限公司

地址：北京市海淀区中关村南大街 9 号理工科技大厦 13 层

网址：[HTTP://WWW.NORTIONTECH.COM](http://www.nortiontech.com)

24 小时服务热线：400-600-8482

E_mail:

sales@nortiontech.com

support@nortiontech.com

感谢您选择 iDrive 智能伺服驱动器，在使用 iDrive 驱动器之前，请先阅读以下内容：

注意事项

- 请使用者不要随意拆装、修理本产品，出现故障时应向我们维修中心咨询；
- 强烈的冲击或震动可能导致产品机身及附件的故障，因此请小心使用和搬运；
- 在因器件烧毁、发热而导致产品不能正常运作时，应迅速切断电源，并与本公司联络；
- 强力压挤或冲击传感器接口，可能导致产品工作异常；
- 本机工作时会散热，因而需要有一定的环境空间以确保空气流通，同时应远离热源；
- 用户在产品维修之前应备份驱动器的有关数据文档，以免维修过程中所造成的数据丢失；
- 本产品使用高电压，请小心使用以避免触电；
- 不正当的安装非本公司产品的软件会导致软、硬件冲突，从而导致系统不能正常工作；
- 操作者须对 PID 控制特别是有关控制参数的调节有一定认识，不适当的调试会导致系统操作不正常甚至烧毁。

操作步骤

为了能使 iDrive 驱动器能正常运行，请按照以下步骤进行操作和调试：

第一步：按照用户手册的指导安装驱动器硬件；

第二步：利用 motionStarter 上位机调试软件进行驱动器的调试和试运行；

第三步：利用以下方法之一进行驱动器的应用程序编程

- 利用 motionLib 函数库编写应用程序；
- 独立运行模式编程；
- 利用驱动器的命令集进行 PLC、HMI 或其它设备的编程；

第四步：根据所设计的应用程序和操作步骤对驱动器进行操作。

目 录

1. 安全须知.....	5
1.1 安全警告.....	5
1.2 注意事项.....	6
2. iDrive 驱动器概述	7
2.1 简介	7
2.2 iDrive 智能驱动器主要特点.....	9
2.3 支持的电机类型	10
2.4 驱动器操作模式	11
2.4.1 RS232 网络模式	12
2.4.2 RS485 网络模式	12
2.4.3 独立运行模式	13
2.4.4 脉冲运行模式	14
2.4.5 模拟信号模式	15
2.4.6 全闭环模式	16
2.5 尺寸	17
2.6 电气特性	18
2.7 iDrive 驱动器型号信息.....	19
3. iDrive 驱动器硬件安装及接线	20
3.1 安装	20
3.2 驱动器接口与连接	21
3.2.1 驱动器工作状态指示器.....	23
3.2.2 输入/输出接口.....	24
3.2.3 地址和通讯设置.....	28
3.2.4 RS232/RS485 通讯接口	29
3.2.5 电源接口	30
3.2.5 电机制动器接口.....	31
3.2.6 编码器接口	31
3.2.7 电机动力线接口.....	32
3.3 第一次上电	33

4. iDrive 的调试	34
4.1 motionStarter 调试软件的安装.....	34
4.2 iDrive 驱动器控制回路介绍.....	34
4.3 用 motionStarter 调试 iDrive.....	37
4.3.1 连接驱动器和上位机	37
4.3.2 motionStrater 主界面	39
4.3.3 系统及电机参数设置.....	41
4.3.4 输出端口设置	44
4.3.4 电流环调试	45
4.3.5. 速度/位置环调试.....	53
4.3.6. 驱动器参数备份.....	62
4.3.7 其它注意事项	62
4.3.8 系统参数复制	62
4.3.9 更新系统参数	63
4.3.10 保存系统参数.....	63
4.3.11 故障诊断	64
4.3.12 下载系统参数.....	65
4.3.13 下载系统程序.....	66
4.3.14 运动程序编程和下载.....	67
5. 使用 motionLib 编写用户程序.....	79
5.1 motionLib 的安装.....	79
5.2 motionLib 的结构.....	80
5.3 motionLib 函数功能说明.....	81
5.4 利用 Visual C++ 进行用户程序的编程.....	85
6. iDrive 命令一览	89
7. 附录.....	90
(1) 故障诊断信息	90

1. 安全须知

在对驱动器进行安装和调试前请先仔细阅读本章节中的信息。这些信息意在您使用本产品时保护您、驱动器及相关的设备的安全。不正确的使用驱动器会导致人生意外伤害或财产损失。只有专业人员才可以安装、调试、操作和维护驱动器，或由一个有资格认证的专业人员授权其他人执行诸如运输、组装、安装、调试和操作驱动器的任务。

在本手册中有以下安全标志：



警告！操作引起人身安全的危险标记，也包括预防这种情况发生的使用说明。



注意！驱动器可能损坏产品或设备的危险标记，也包括避免此情形发生的使用说明



注意！指示该区域对静电敏感，要求在静电保护环境中处理。

1.1 安全警告



警告！驱动器电压可能引起电击，电源打开的情况下，请不要直接接触带电部分。



警告！为了避免电弧放电引起的危险，当电源打开时请不要连接或断开驱动器与电源的接线以及插拔驱动器上的接口。



警告！驱动器在运行期间表面可能发热。



警告！在驱动器运行期间，被控制的电机在旋转运动并带动机械系统运行，请远离所有运动部件，免受伤害。

1.2 注意事项



注意！连接驱动器的供电电源必须遵守本手册中的参数规定。



注意！故障维修与服务只有经过授权许可的专业人员才可以执行。



注意！驱动器包含一些静电敏感元器件，如果处理不当会被损坏。因此驱动器应该在静电保护环境中从包装中取出。

为了防止静电引起的损坏，请避免接触绝缘材料，如合成纤维或塑胶制品。为了释放静电，请注意保持驱动器的良好接地。

2. iDrive 驱动器概述

2.1 简介

iDrive 是一款基于DSP及FPGA 技术的全数字智能伺服驱动器，iDrive集成了多类电机驱动与运动控制功能于一体。iDrive为诺信泰自主知识产权和专利权的伺服驱动产品，我们的目标是为客户提供一个高性能、高可靠性、功能丰富、低成本的运动控制解决方案。

iDrive智能伺服驱动器适用于控制驱动永磁直流无刷伺服电机、永磁交流伺服电机、直流有刷电机等电机。iDrive 智能伺服驱动器集运动控制、驱动和集成的PLC 功能于一个紧凑的单元中，无需外部运动控制器，即可实现运动控制功能。iDrive智能驱动器可执行位置、速度、单轴、多轴或独立工作模式等多种功能。

iDrive智能伺服驱动器具有很高的动态响应性能，能够在高速的应用中保证运动控制的精度。iDrive智能伺服驱动器可以通过RS232或RS485通讯组成运动控制网络，或者是由事先存储在驱动器Flash中的运动程序执行运动控制任务。在独立运行模式，运动任务可以由输入/输出接口的状态来触发，或者由上位机的命令来触发。iDrive还可以和通用伺服一样接收脉冲和模拟信号命令进行电机的控制。iDrive的多种控制模式为用户组建运动控制系统提供多方位的解决方案。

iDrive智能伺服驱动器提供了单轴和双轴两种模块，图1分别显示了单双轴iDrive驱动器的接口示意，具体接口请参考以下相关章节。



图1. iDrive智能伺服驱动器单轴和双轴模块

运用 iDrive 智能驱动器组成运动控制系统：

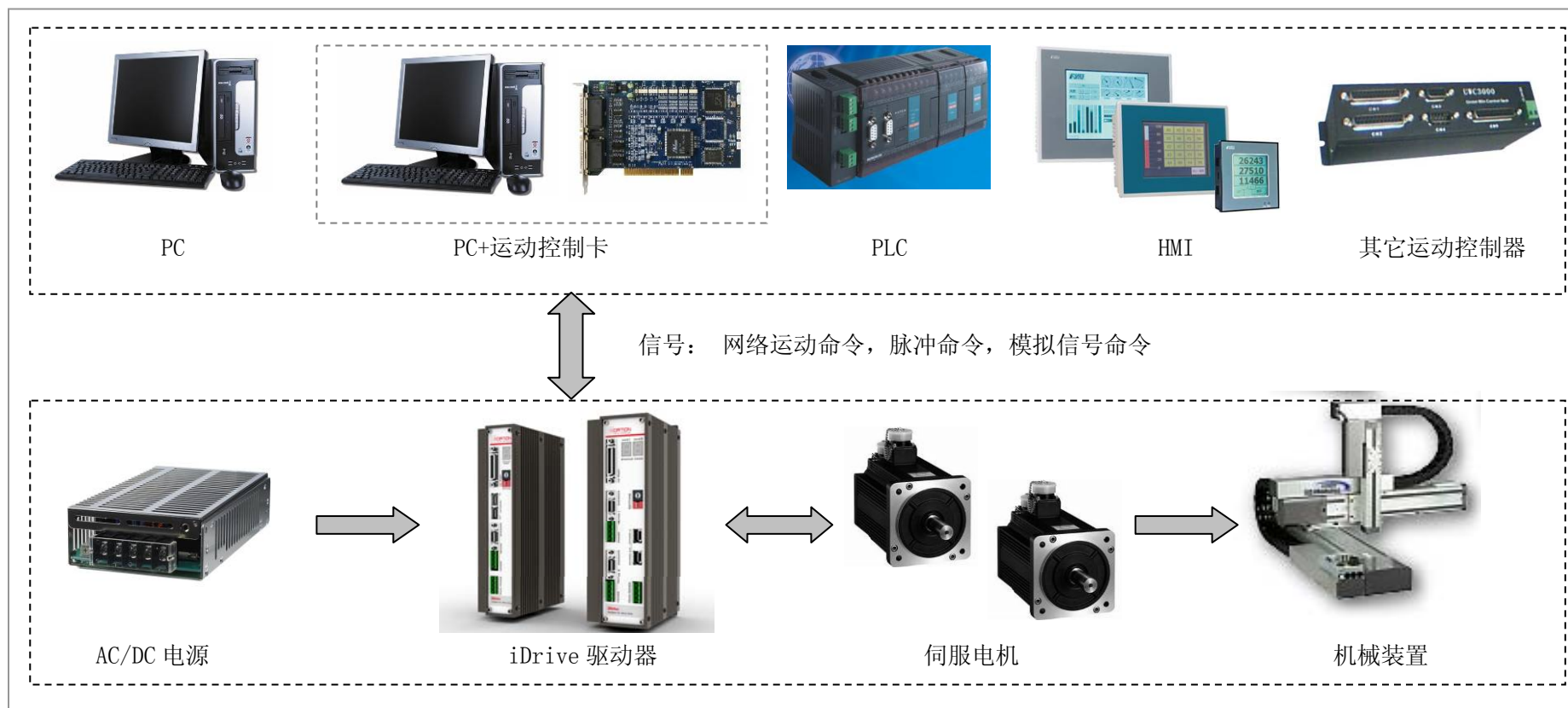


图 2. 用 iDrive 组成运动控制系统

2.2 iDrive 智能驱动器主要特点

iDrive 智能驱动器具有以下主要特点：

- 全数字化分布式运动控制和驱动一体化系统
- 控制轴数为单轴或双轴
- 适用于永磁直流无刷(有刷)伺服电机、永磁交流伺服电机以及直线电机
- 模块化结构，可以替换不同的驱动模块而组成不同功率等级的智能驱动器
- 电流环控制频率：20KHz
- 位置环和速度环控制频率：4KHz
- 控制模式
 - 网络控制模式
 - 带有运动程序存储功能的独立运行模式
 - iDrive 做为驱动器应用的脉冲模式
 - iDrive 做为驱动器应用的模拟信号命令模式
 - 全闭环模式
- 网络功能
 - RS232
 - RS485（连接 31 台控制器并可扩展）
- 两轴位置、速度同步功能
- 轨迹规划功能
 - S 速度曲线（限制加加速度功能）
- 供电电压
 - 逻辑电路 24V
 - 驱动电压 24V-100V
- 电流
 - 具有多个电流等级，最大额定值 16A， 峰值 28A
- 输入/输出接口
 - 单轴模块提供 5 个光电隔离输入、2 个差分输入、3 个差分输出、3 个光电隔离输出和 2 个模拟输入
 - 双轴模块提供 12 个光电隔离输入、6 个光电隔离输出和 2 个模拟输入
- 带有用于系统配置、调试及图形化编程的 motionStarter 软件，以及用于应用程序设计的动态函数库 motionLib
- 具有过流、过压、欠压、过载、位置超差、温度超限等保护功能，网络故障诊断及现场故障诊断等功能

iDrive 智能驱动器具有以下优点：

- 更高的可靠性

iDrive控制器和驱动器一体化的结构不但减少了连接电缆的数量，同时也提高了系统的可靠性，降低外界干扰对系统的影响。

- 更大的灵活性

分布式结构给系统配置和机器设计灵活性带来极大的方便，增加或减少控制轴都能很容易灵活的完成。控制器本地化安装减少了对控制柜的要求，而控制柜将不再是限制系统灵活性的瓶颈。使用分布式结构，能通过组合不同的模块，很容易地构造极为个性化的运动控制解决方案。

- 更快的动态响应

控制器和驱动器一体化的结构保证了各个控制环较高的采样和控制频率，较高的控制频率保证了系统有较高的动态响应。从而能保证电机在高速运动状态下系统的运动控制精度。

- 更具节约潜力

分布式运动控制驱动单元通过总线进行连接通讯，一条总线可同时连接最高可达31个控制器。对于大型系统的安装，采用直流电源线 and 通讯总线结构，可以显著减少所需电缆数量。另外，简单的系统连接可保证在安装和维修方面节约大量的成本。

- 突出的特性

采用界面友好的上位机应用程序进行工程调试，可实现快速和简单的调试、参数配置以及诊断。统一的直流电源总线、网络功能和紧凑结构使工程量显著减少，而故障排查更为方便。图形化的运动编程及独立运行模式使系统的应用更方便。控制环频率的提高使得机器在高速运行中同样能保证很好的动态精度和响应速度。

2.3 支持的电机类型

iDrive 智能驱动器支持的电机类型有永磁交流伺服电机、永磁直流无刷伺服电机、直流有刷电机以及直线电机。详细介绍如下：

1) 永磁交流伺服电机，电机带增量式编码器，有霍尔信号或无霍尔信号均可连接到 iDrive 驱动器，系统如下：

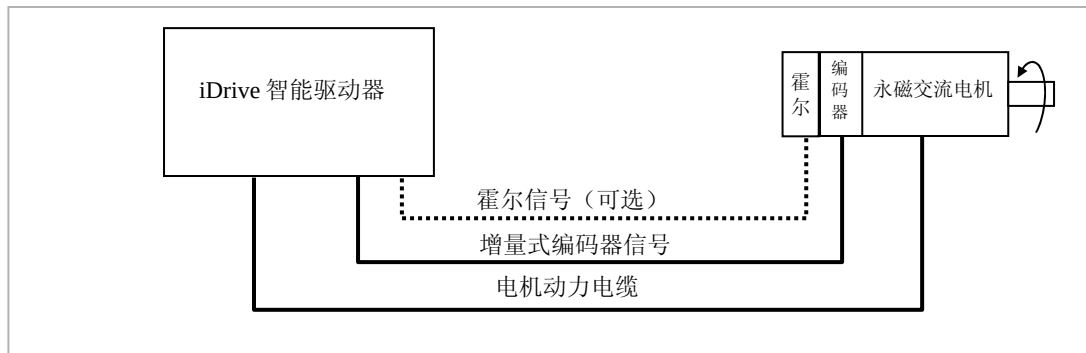


图 3. iDrive 智能驱动器连接永磁交流伺服电机

2) 永磁直流无刷伺服电机，电机带增量式编码器和霍尔传感器，系统如下：

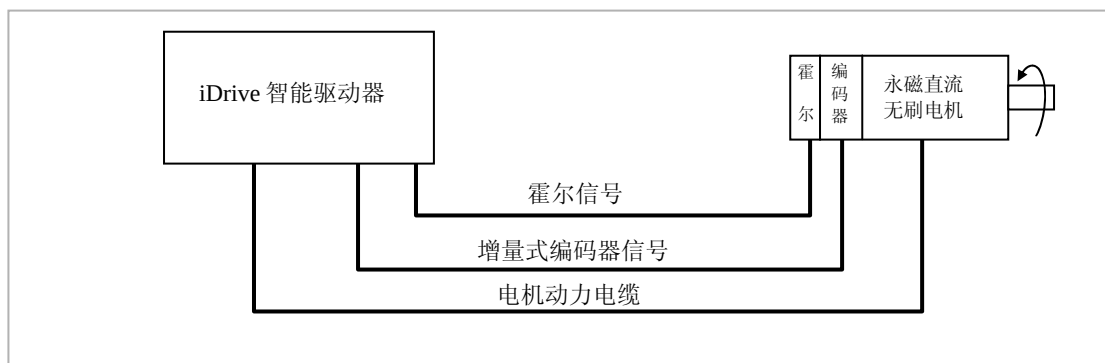


图 4. iDrive 智能驱动器连接永磁直流无刷伺服电机

3) 直流有刷电机，电机带增量式编码器，系统如下：

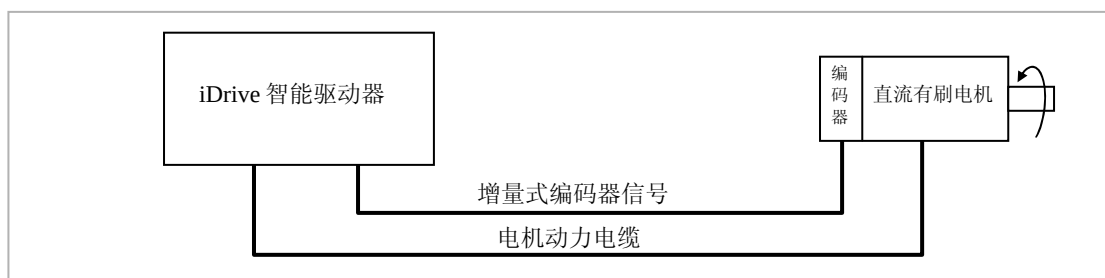


图 5. iDrive 智能驱动器连接直流有刷电机

4) 直线电机，电机带直线光栅尺，霍尔信号可选，系统如下：

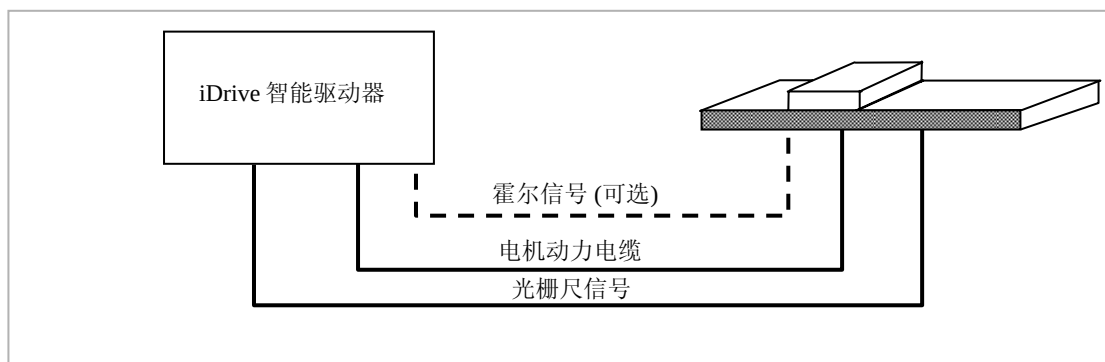


图 6. iDrive 智能驱动器连接直线电机

2.4 驱动器操作模式

iDrive 智能伺服驱动器支持的操作运行模式有：

- 网络运行模式
 - RS232 网络模式
 - RS485 网络模式
- 独立运行模式

- iDrive 作为通用驱动器使用的脉冲运行模式
- iDrive 作为通用驱动器使用的模拟信号命令模式
- iDrive 应用于全闭环控制模式

各操作模式介绍如下：

2.4.1 RS232 网络模式

当 iDrive 运行于 RS232 网络模式时，用户可以利用 RS232 通讯线连接到计算机进行系统调试和实时控制，并可利用所提供的函数库或命令集开发自己的上位机应用程序。上位机可以是 PC、PLC 或 HMI，一个 RS232 接口能连接一台驱动器（单轴或双轴都可以连接），如果上位机有多个 RS232 接口，也可以连接多台 iDrive 伺服驱动器。在 RS232 网络模式中，上位机随时都可以得到个 I/O 的状态并可控制其中的输出接口的状态，各个轴之间可以同步协调工作。图 7 显示了用 RS232 组建的 iDrive 运动控制系统。

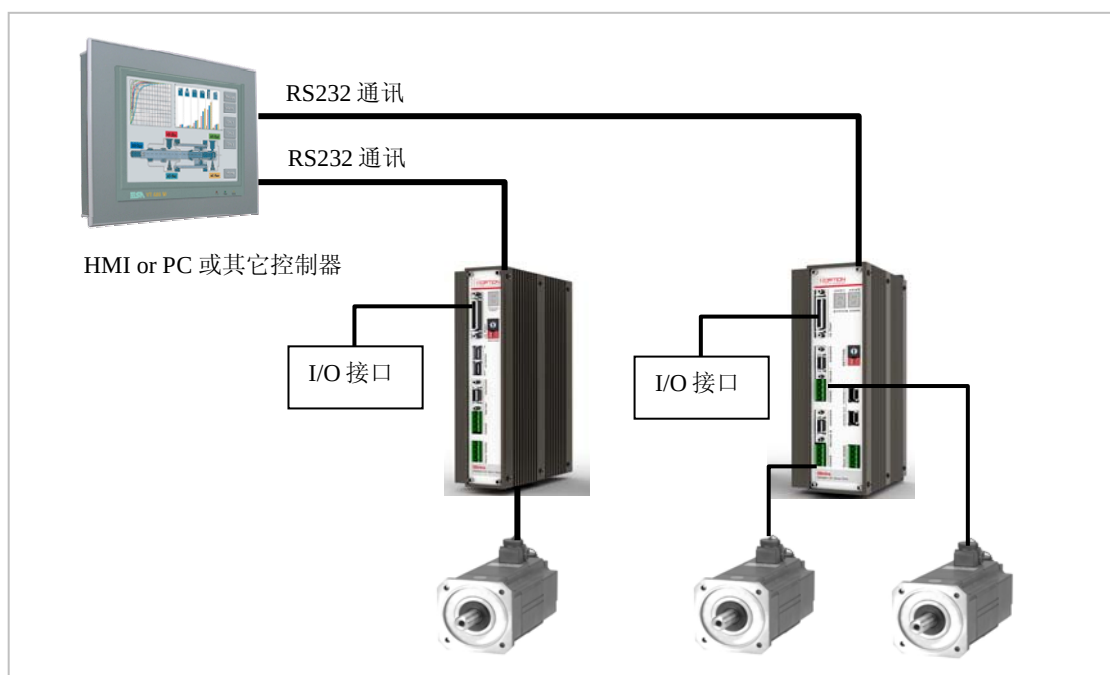


图 7. iDrive 应用于 RS232 网络模式

2.4.2 RS485 网络模式

iDrive 可以利用 RS485 网络组建分布式多轴运动控制系统，在分布式多轴运动控制系统中，上位机既能给 RS485 网络中的各个轴发送数据，也能发送命令执行每个轴中板载 Flash 中的运动程序，或通过板上的输入/输出接口做协同运动。同时，上位机也能对各个 I/O 接口进行采集和控制。图 8 显示了用 RS485 组建的 iDrive 分布式运动控制系统。运行于 RS485 网络模式时，用户可以利用 RS485 通讯连接到计算机进行系统调试。

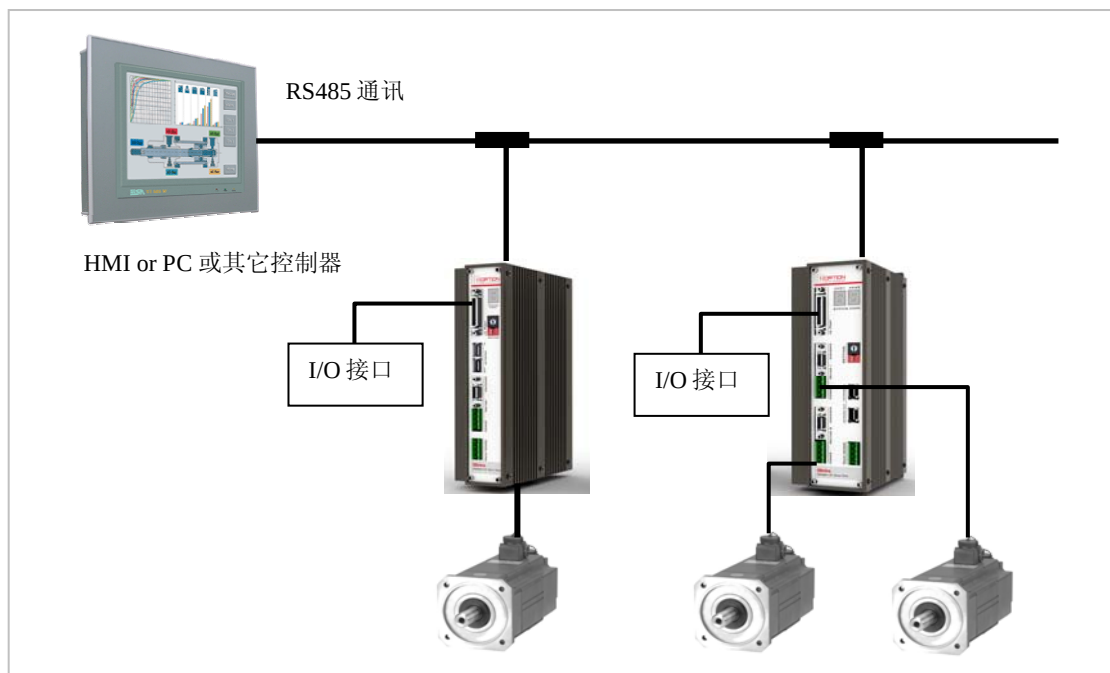


图 8. iDrive 应用于 RS485 网络模式

2.4.3 独立运行模式

iDrive 智能伺服驱动器能以独立的模式进行运行，此时电机受到驱动器 Flash 中的运动程序所控制。通过 motionStarter 软件平台设置电机、驱动器参数和设计运动控制程序，编译下载到驱动器上的 Flash 中，然后设置驱动器为独立运行（Standalone）方式重新上电后，电机将按照 Flash 中的运动顺序运行。独立运行模式下可以由输入/输出口的状态来触发和终止运动程序的运行（也可以手动控制）。图 9 显示了独立运行模式的系统示意图。

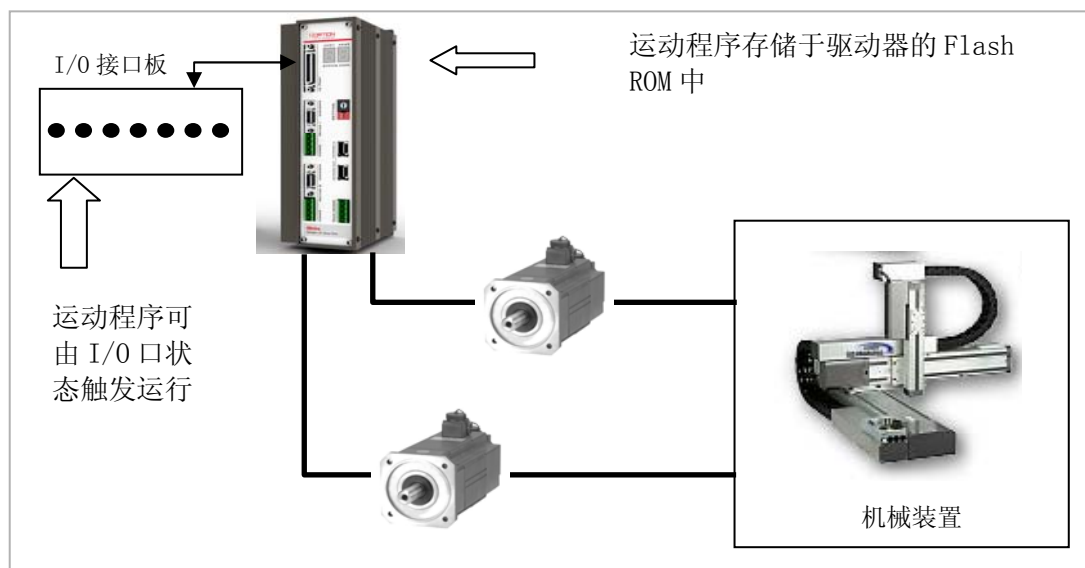


图 9. iDrive 应用于独立运行模式

2.4.4 脉冲运行模式

iDrive 智能伺服驱动器作为通用驱动器使用时，能接受外部运动控制器（运动控制卡或 PLC）发送来的脉冲控制信号作为位置或速度运动命令。并且可以由软件设置不同的电子齿轮比例值。其操作模式就如同传统的基于脉冲控制模式的驱动器一样。图 10 给出了工作于脉冲/方向模式是系统的连接方式，图 11 显示了用于脉冲/方向运行模式时的 iDrive 系统。

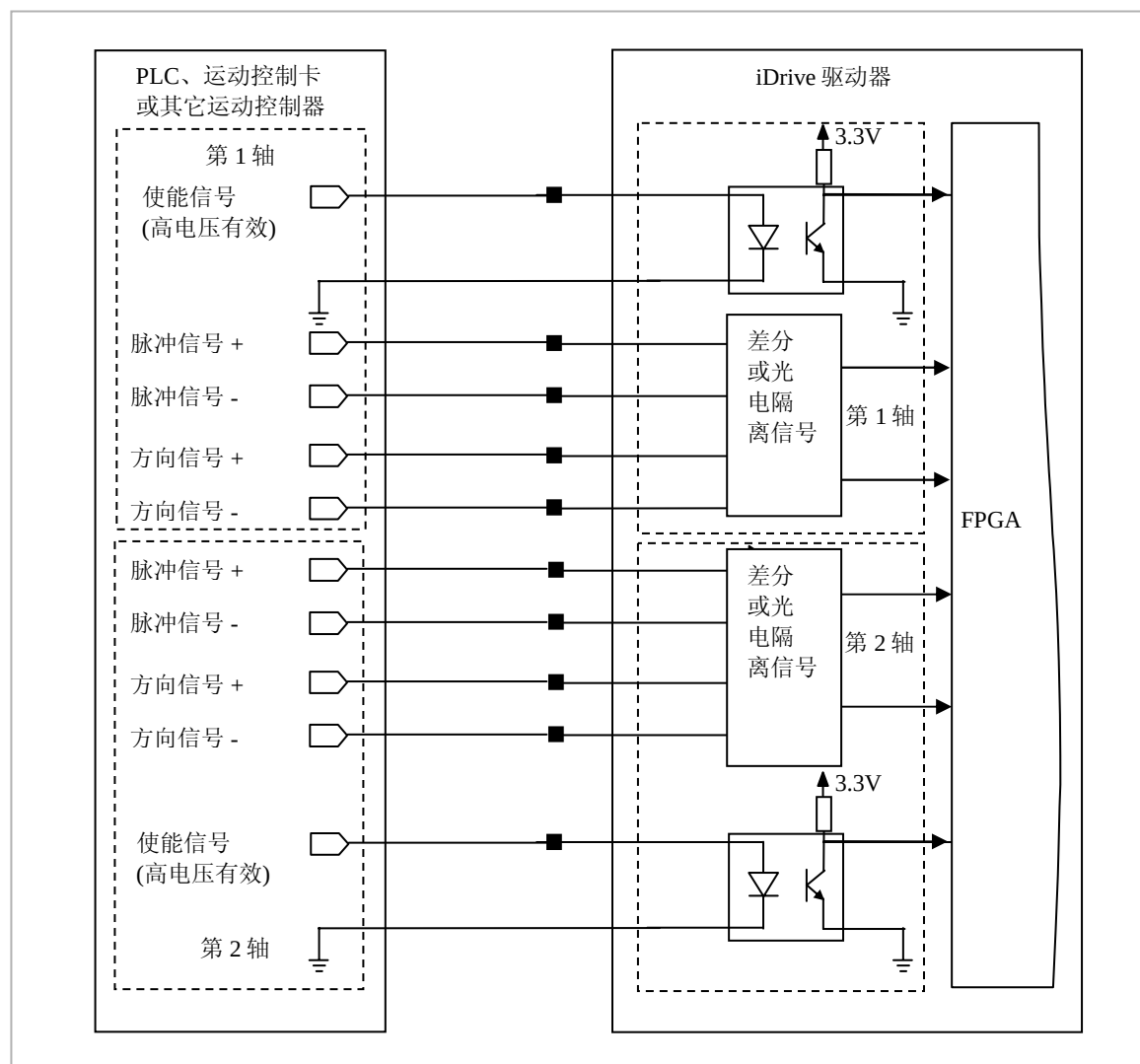


图 10. iDrive 作为驱动器应用于脉冲/方向模式时的连接方式示意图

当 iDrive 驱动器工作在脉冲/方向模式时，驱动器输入的脉冲和方向信号可以是差分信号也可以是光电隔离信号，如图 10 所示。驱动器差分信号的连接请参见下一节输入/输出接口的连接。

当驱动器工作于脉冲模式时，除了脉冲/方向模式，我们还可以提供其他方式的脉冲工作模式。

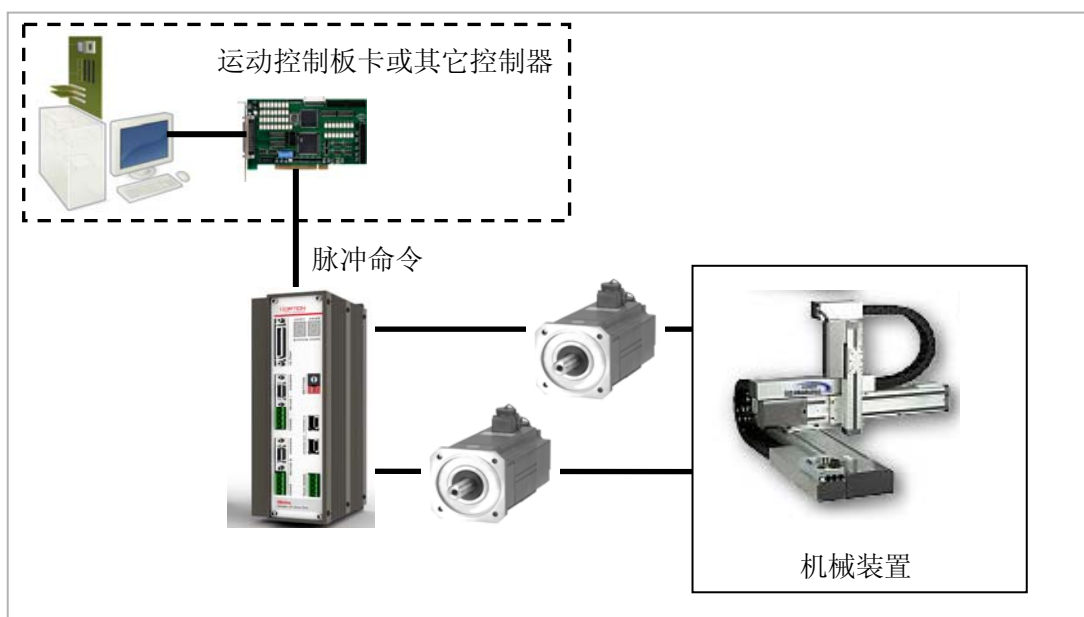


图 11. iDrive 应用于脉冲/方向模式

2.4.5 模拟信号模式

iDrive 智能伺服驱动器作为通用驱动器使用时，能接受外部运动控制器（运动控制卡或 PLC）发送来模拟信号命令进行伺服系统的控制。当 iDrive 驱动器工作在模拟信号命令模式时，其系统示意图如图 12 所示。驱动器模拟信号的连接请参见下一节输入/输出接口的连接。

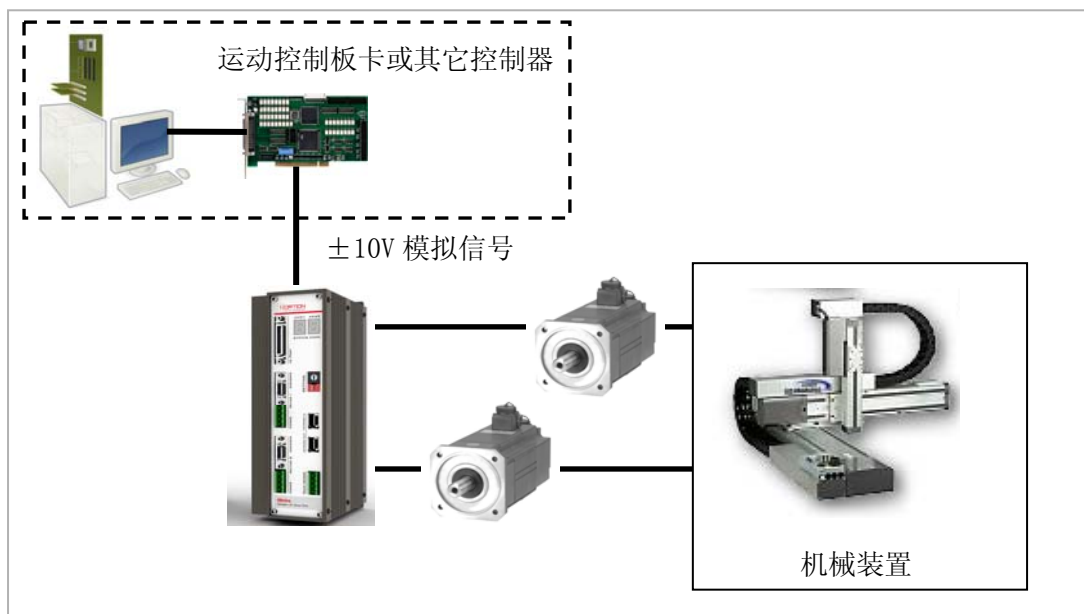


图 12. iDrive 应用于模拟信号命令模式

2.4.6 全闭环模式

iDrive 智能伺服驱动器除了可以接收电机轴端的编码器信号之外，还可以通过 I/O 口接收安装于机械轴上的第二编码器信号，工作于全闭环工作模式，其系统示意图如图 13 所示。

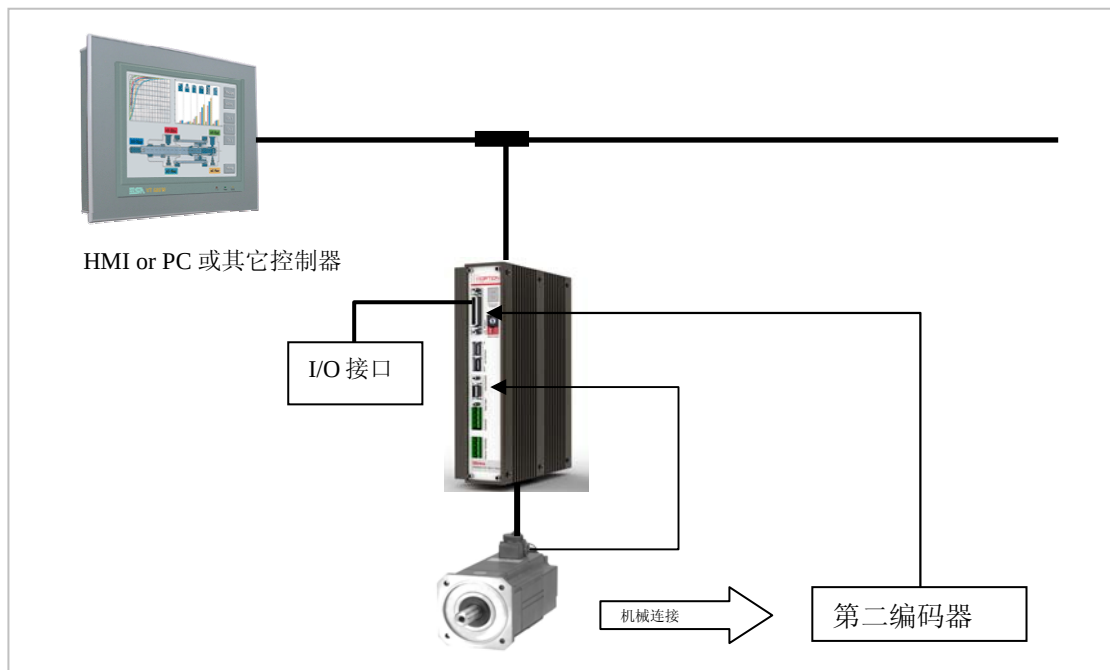
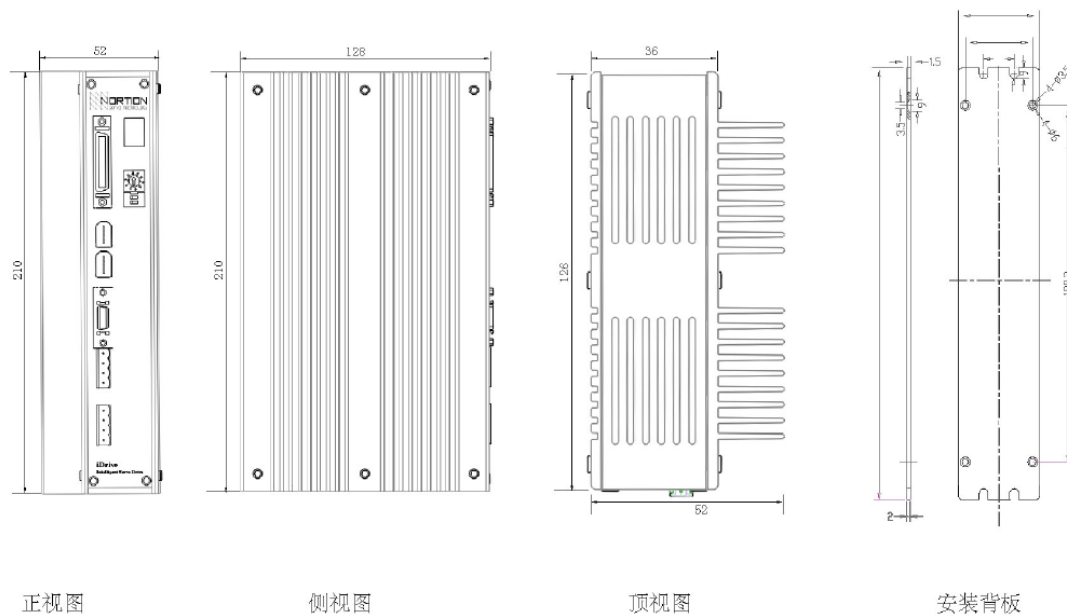


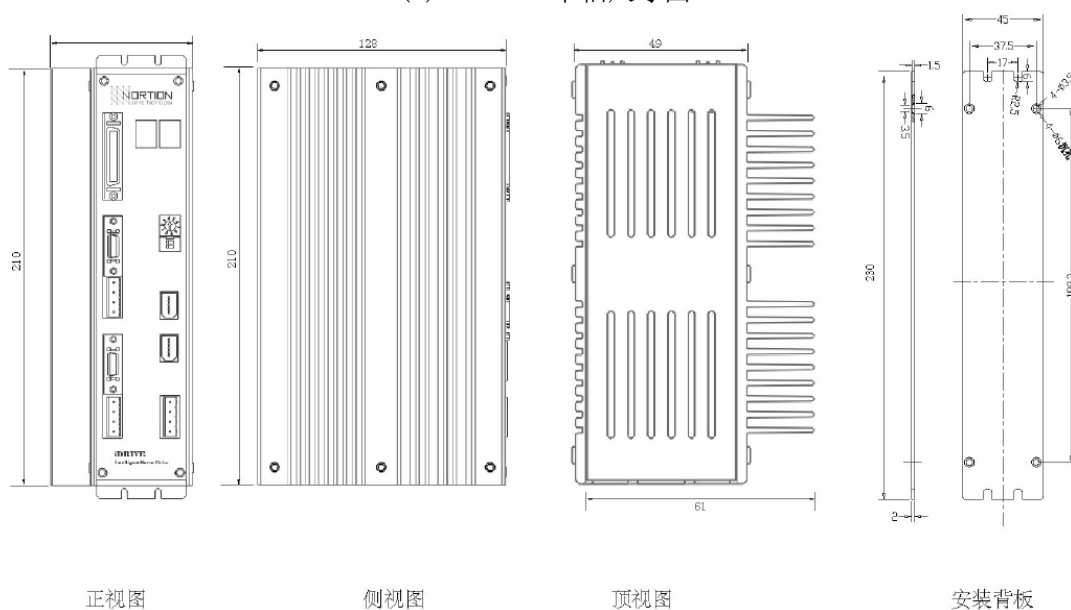
图 13. iDrive 应用于全闭环模式

2.5 尺寸

iDrive 驱动器尺寸图:



(1) iDrive 单轴尺寸图



(2) iDrive 双轴尺寸图

图 14. iDrive 驱动器尺寸

2.6 电气特性

表 1. 运行条件

	最小值	典型值	最大值	单位
环境温度	0	25	+40	°C
壳体温度(金属表面)	0	25	+60	°C
环境湿度(无)	–	–	90	%RH
静电保护	± 25	± 30		KV

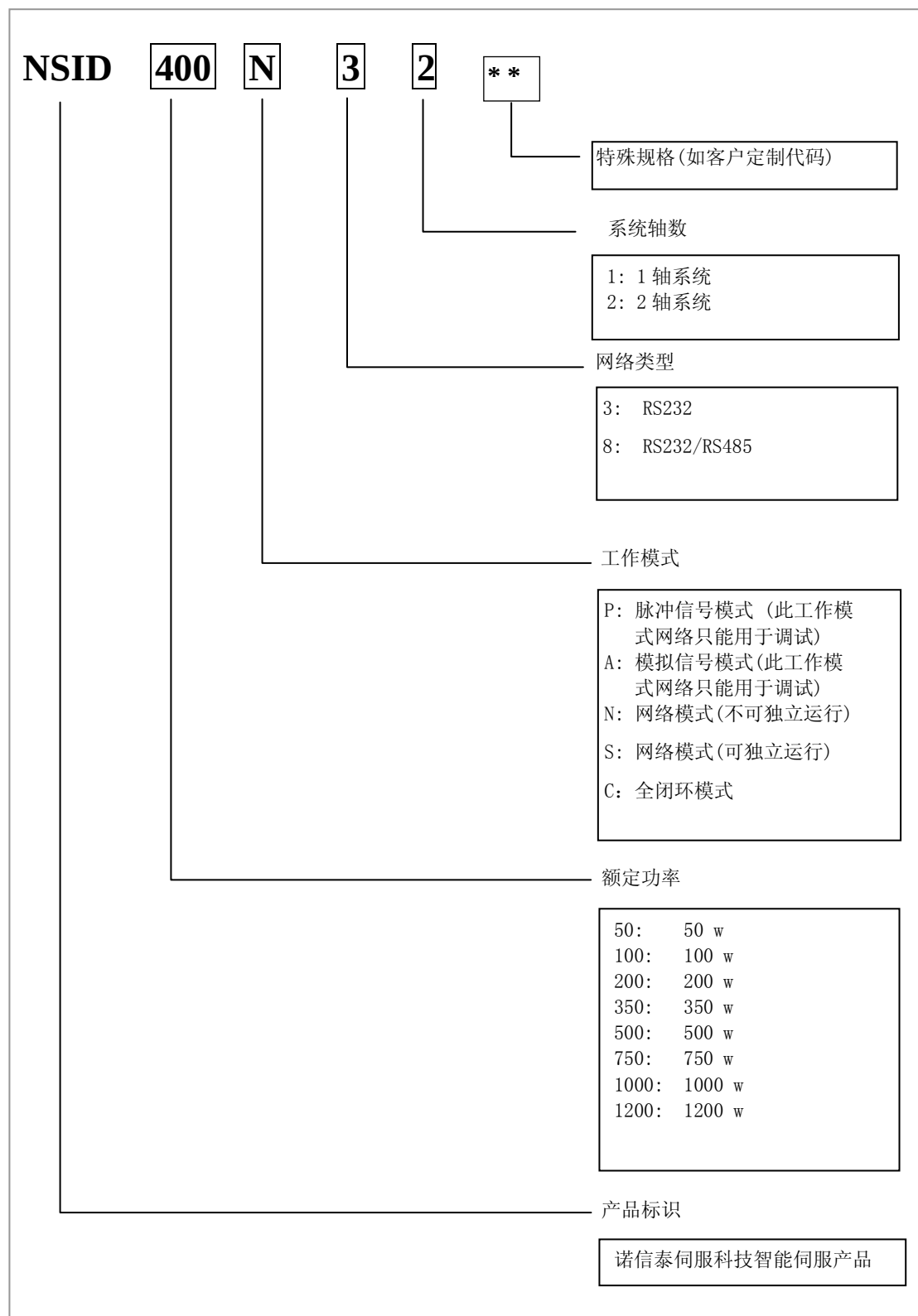
表 2. 逻辑电源输入

	最小值	典型值	最大值	单位
电源电压	12	24	30	V _{DC}
电源电流	–	150	500	mA

表 3. 驱动电源输入

	最小值	典型值	最大值	单位
电源电压	24	–	100	V _{DC}
电源电流	–	12	16	A

2.7 iDrive 驱动器型号信息



更详细的选型信息请参考iDrive选型手册。

3. iDrive 驱动器硬件安装及接线

3.1 安装

■ 冷却要求:

iDrive 驱动器设计为自然冷却，可以水平安装放机柜中（见图15），或垂直安装在机箱中。

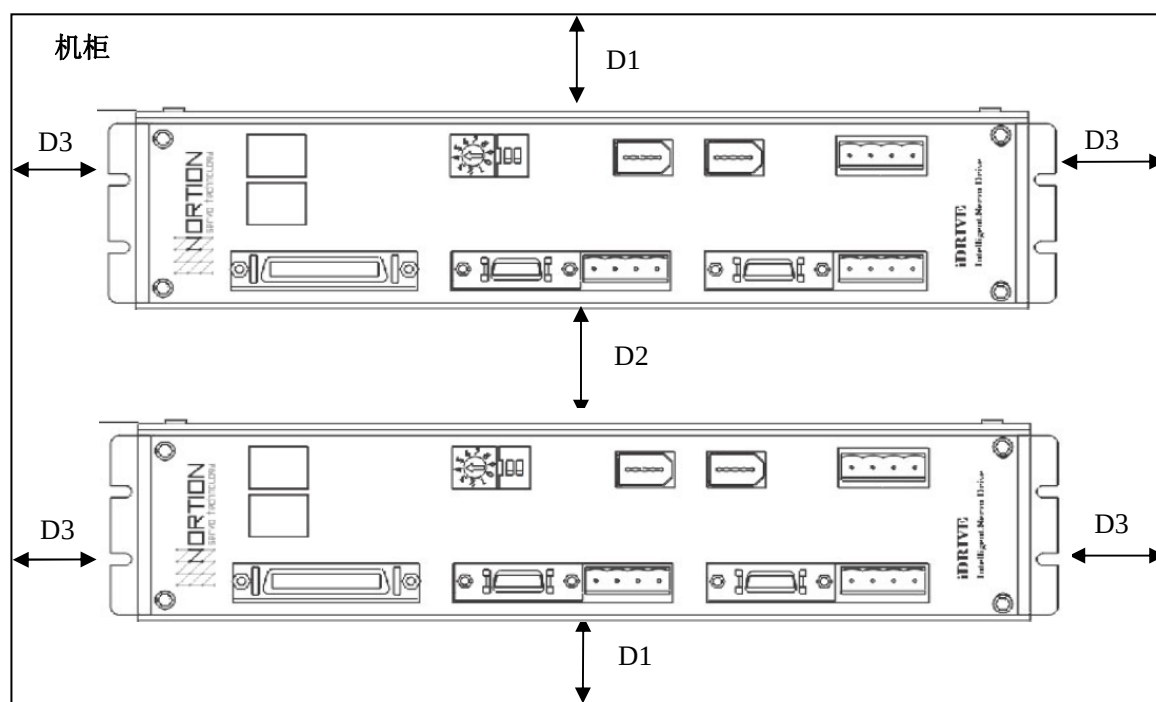


图 15. iDrive 安装尺寸图

驱动器与周围柜壁以及驱动器之间的距离如D1, D2 和 D3，保证空气自由循环。

表4. 驱动器安装冷却距离要求

冷却距离需求	
D1	> 25mm
D2	> 60mm
D3	> 25mm

■ 接线要求

D1、D2、D3安装距离必须能保证电缆线连接到驱动器上。

表5. 接线要求

接线要求	
D1	> 120mm
D2	> 100mm
D3	> 25mm

3.2 驱动器接口与连接

iDrive 驱动器的接口都在驱动器的正面，如图 16 所示。每个接口的功能和电缆连接方式在以下的章节中做详细的介绍。



J1: 驱动器工作状态指示;
J3: 地址和通讯设置;
J6: 1 号电机编码器接口;
J8: 2 号电机编码器接口;
J10: 电源接口

J2: 输入/输出接口;
J4, J5: 通讯接口;
J7: 1 号电机动力电缆接口;
J9: 2 号电机动力电缆接口

图 16. 驱动器接口示意图

图 17 和图 18 分别给出了 iDrive 单轴和双轴驱动器的接口说明示意图。

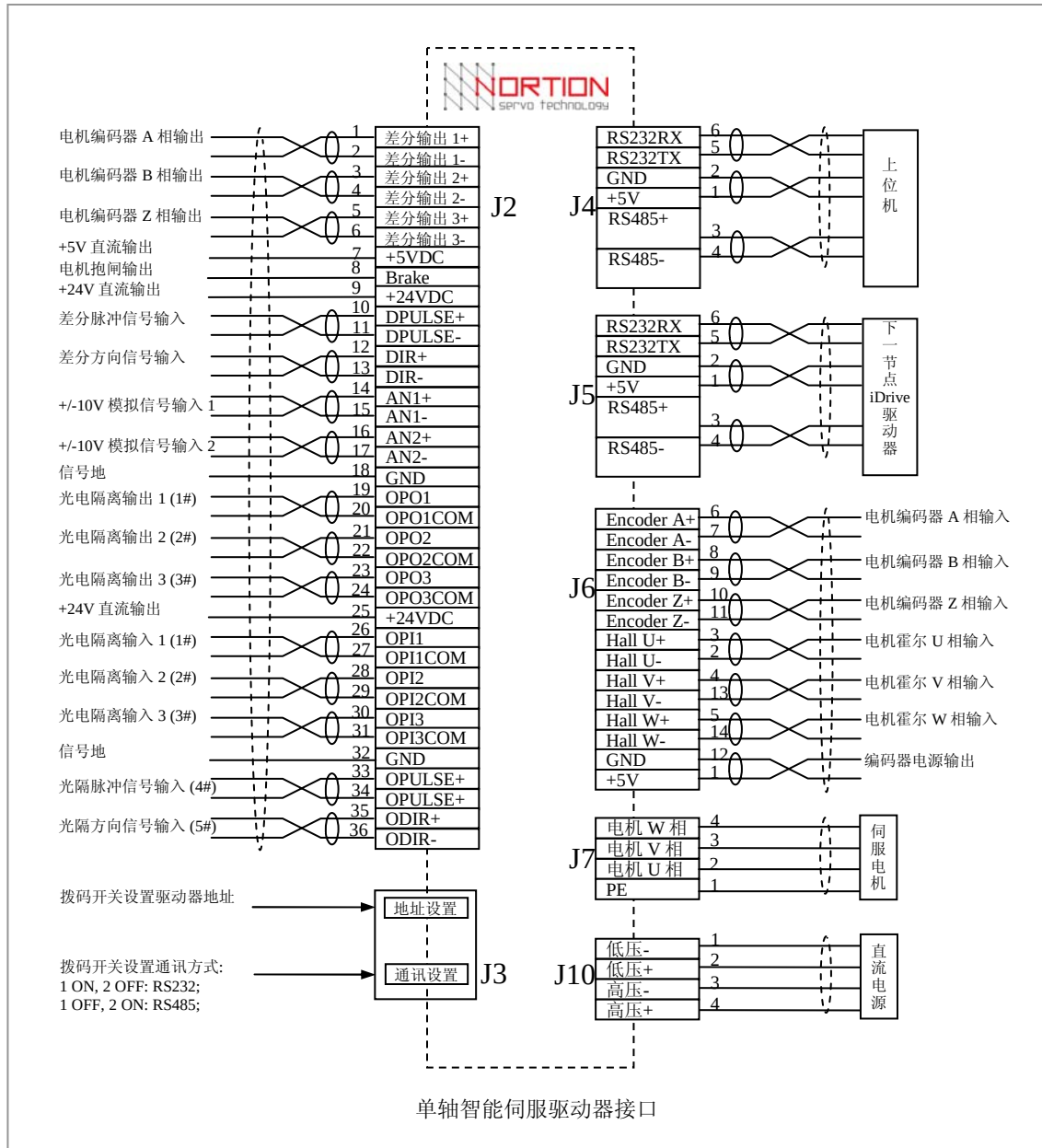


图 17. iDrive 单轴轴驱动器接口说明

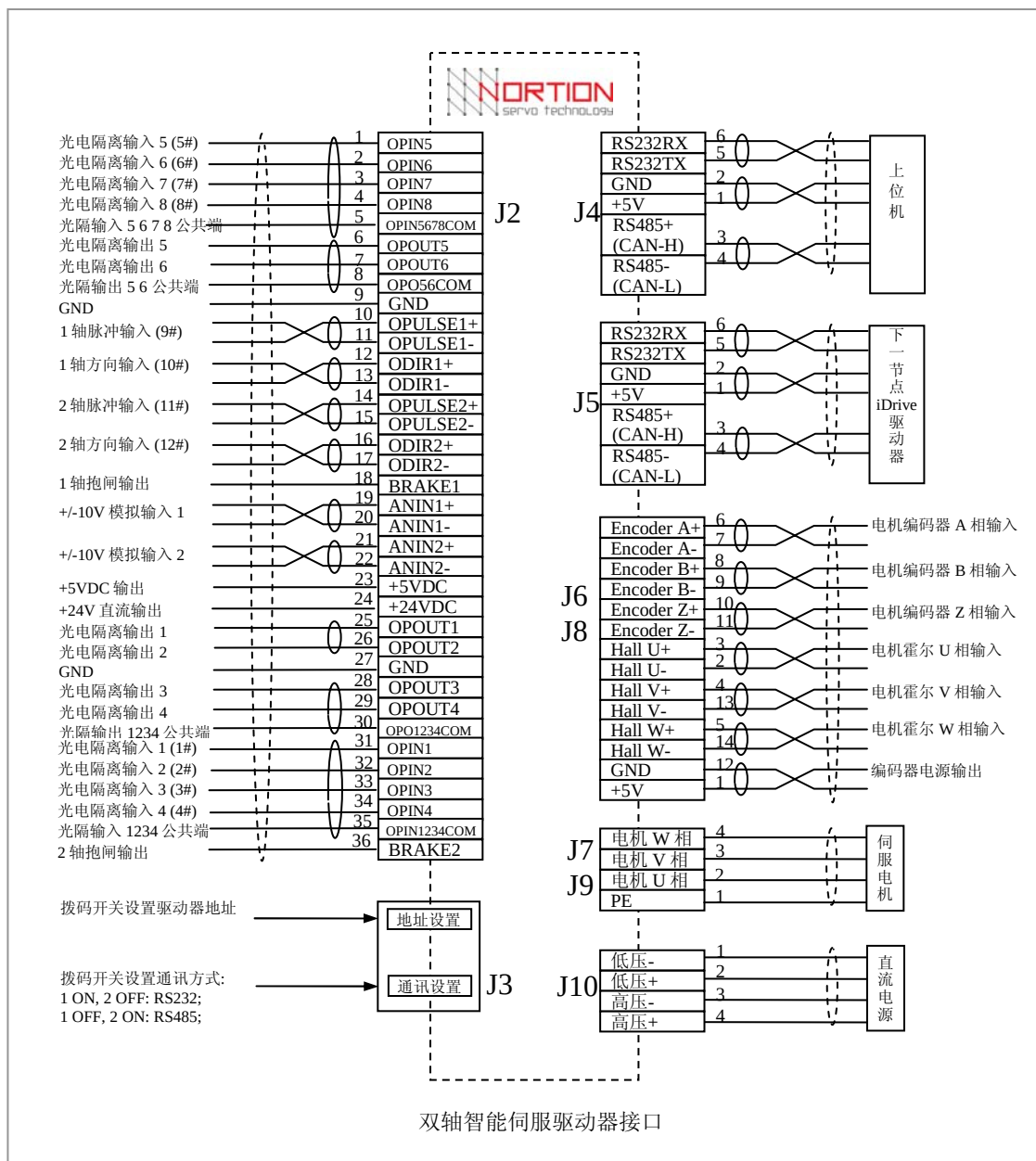


图 18. iDrive 双轴驱动器接口说明

3.2.1 驱动器工作状态指示器

J1 为驱动器工作状态指示器，可以用于驱动器现场故障诊断。如下图所示：

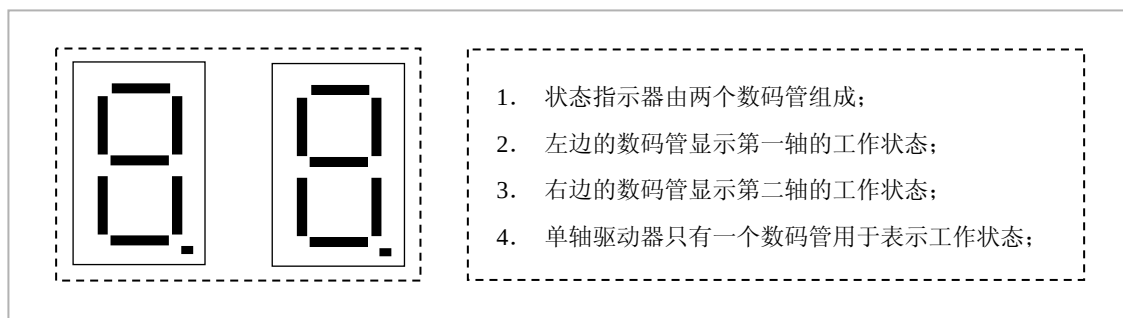


图 19. 驱动器工作状态指示器

表 6 给出了故障代码一览表。

表 6. iDrive 故障代码和故障内容一览表

代码	故障内容
8	系统没有故障
0	系统启动未完成
1	参数检查错误
2	A/D 转换故障
3	霍尔信号错误
4	跟踪超限错误
5	编码器故障
6	速度超限错误
7	操作超时警告
9	驱动部分欠压
11	平均电流错误
A	A/D 转换超时错误
C	制动器短路
E	放大器短路
F	致命系统错误
H	外部电压超载 (5VDC)
L	过压
n	驱动器启动顺序错误
P	FPGA 返回故障
U	A/D 转换偏置故障
	通信超时
	温度警告
	温度故障

3.2.2 输入/输出接口

J2 号接口是输入/输出信号的接口，用于驱动器和外部设备相连。提供的输入/输出接口为一个 36 针的 SCSI 接口，示意如下图：

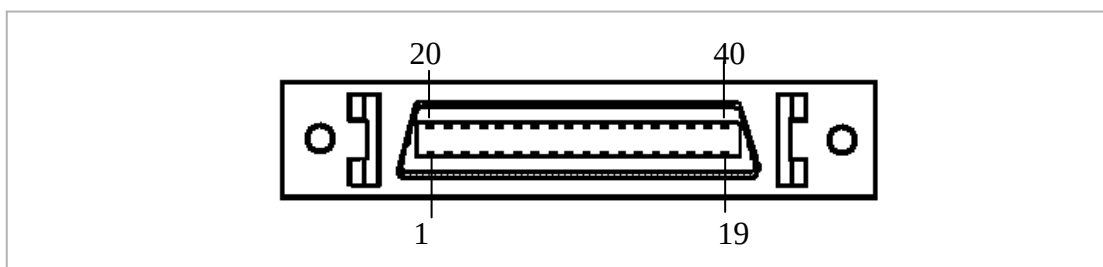


图 20. 输入/输出接口示意图

iDrive 单轴模块共有 5 个光电隔离输入、2 个差分输入、3 个差分输出、3 个光电隔离输出以及 2 个 $\pm 10V$ 模拟输入。iDrive 双轴模块共有 12 个光电隔离输入、6 个光电隔离输出以及 2 个 $\pm 10V$ 模拟输入。

输入/输出详细说明见表7和表8:

表7. 单轴iDrive输入/输出接口说明

针号	名称	意义	
1	Differential Out1(+)	编码器 A 相输出信号	
2	Differential Out1(-)		
3	Differential Out2(+)	编码器 B 相输出信号	
4	Differential Out2(-)		
5	Differential Out3(+)	编码器 C 相输出信号	
6	Differential Out3(-)		
7	+5VDC	+5VDC 输出	
8	BRAKE	电机抱闸输出	
9	+24VDC	+24VDC 输出	
10	Diff Pulse +	工作于脉冲方向模式的差分脉冲输入(输入口 6#)	
11	Diff Pulse -		
12	Diff Dir +	工作于脉冲方向模式的差分方向输入(输入口 7#)	
13	Diff Dir -		
14	Analog-In1(+)	模拟信号输入通道 1#	
15	Analog-In1(-)		
16	Analog-In2(+)	模拟信号输入通道 2#	
17	Analog-In2(-)		
18	GND	GND	
19	OPTO OUT 1	光电隔离输出 1#	
20	OPTO OUT 1 COM	光电隔离输出 1# COM 端	
21	OPTO OUT 2	光电隔离输出 2#	
22	OPTO OUT 2 COM	光电隔离输出 2# COM 端	
23	OPTO OUT 3	光电隔离输出 3#	
24	OPTO OUT 3 COM	光电隔离输出 3# COM 端	
25	+24VDC	+24VDC 输出	
26	OPTO IN 1	光电隔离输入 1#	
27	OPTO IN 1 COM	光电隔离输入 1# COM 端	
28	OPTO IN 2	光电隔离输入 2#	
29	OPTO IN 2 COM	光电隔离输入 2# COM 端	
30	OPTO IN 3	光电隔离输入 3#	
31	OPTO IN 3 COM	光电隔离输入 3# COM 端	
32	GND	GND	
33	OPTO IN PULSE	光电隔离输入 4#	可以复用为脉冲模式 脉冲输入
34	OPTO IN PULSE COM	光电隔离输入 4# COM 端	
35	OPTO IN DIR	光电隔离输入 5#	可以复用为脉冲模式 方向输入
36	OPTO IN DIR COM	光电隔离输入 5# COM 端	

表8. 双轴iDrive输入/输出接口说明

针号	名称	意义
1	OPTO IN 5	光电隔离输入 5#
2	OPTO IN 6	光电隔离输入 6#
3	OPTO IN 7	光电隔离输入 7#

4	OPTO IN 8	光电隔离输入 8#	
5	OPTO IN 5 6 7 8 COM	光电隔离输入 5 6 7 8 公共端	
6	OPTO OUT 5	光电隔离输出 5#	
7	OPTO OUT 6	光电隔离输出 6#	
8	OPTO OUT 5 6 COM	光电隔离输出 5 6 公共端	
9	GND	GND	
10	OPTO Pulse 1 IN +	光电隔离输入 9#	可以复用为 1 轴脉冲模式脉冲输入
11	OPTO Pulse 1 IN -	光电隔离输入 9# COM 端	
12	OPTO Dir 1 IN +	光电隔离输入 10#	可以复用为 1 轴脉冲模式方向输入
13	OPTO Dir 1 IN -	光电隔离输入 10# COM 端	
14	OPTO Pulse 2 IN +	光电隔离输入 11#	可以复用为 2 轴脉冲模式脉冲输入
15	OPTO Pulse 2 IN -	光电隔离输入 11# COM 端	
16	OPTO Dir 2 IN +	光电隔离输入 12#	可以复用为 2 轴脉冲模式方向输入
17	OPTO Dir 2 IN -	光电隔离输入 12# COM 端	
18	BRAKE1	1 轴电机抱闸输出	
19	Analog-In1(+)	模拟信号输入通道 1#	
20	Analog-In1(-)		
21	Analog-In2(+)	模拟信号输入通道 2#	
22	Analog-In2(-)		
23	+5VDC	+5VDC 输出	
24	+24VDC	+24VDC 输出	
25	OPTO OUT 1	光电隔离输出 1#	
26	OPTO OUT 2	光电隔离输出 2#	
27	GND	GND	
28	OPTO OUT 3	光电隔离输出 3#	
29	OPTO OUT 4	光电隔离输出 4#	
30	OPTO OUT 1 2 3 4 COM	光电隔离输出 1 2 3 4 公共端	
31	OPTO IN 1	OPTO IN 1#	
32	OPTO IN 2	OPTO IN 2#	
33	OPTO IN 3	OPTO IN 3#	
34	OPTO IN 4	OPTO IN 4#	
35	OPTO IN 1 2 3 4 COM	OPTO IN 1 2 3 4 COM	
36	BRAKE2	2 轴电机抱闸输出	

图 21 说明了 I/O 接口焊盘编号的定义，将 I/O 接口插到驱动器上，针脚焊盘编号如下图所示。

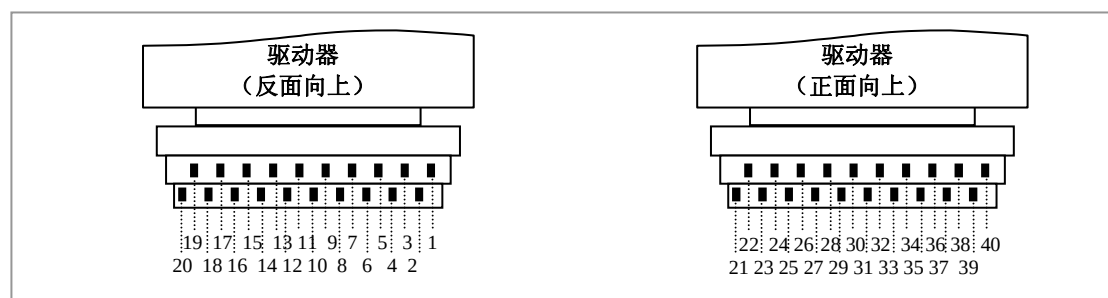


图 21. 输入/输出接口焊盘定义示意图

下面分别说明差分输入、差分输出、光电隔离输入、光电隔离输出和模拟输入的接线方式。

差分输入连接方法如图 22 所示：

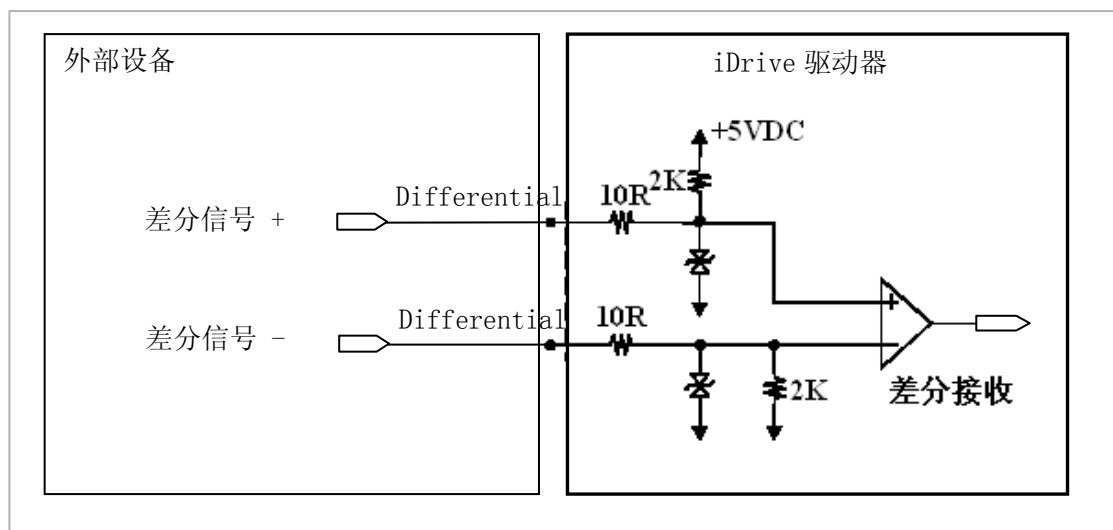


图 22. 差分输入接线方式

差分输出连接方法如图 23 所示。

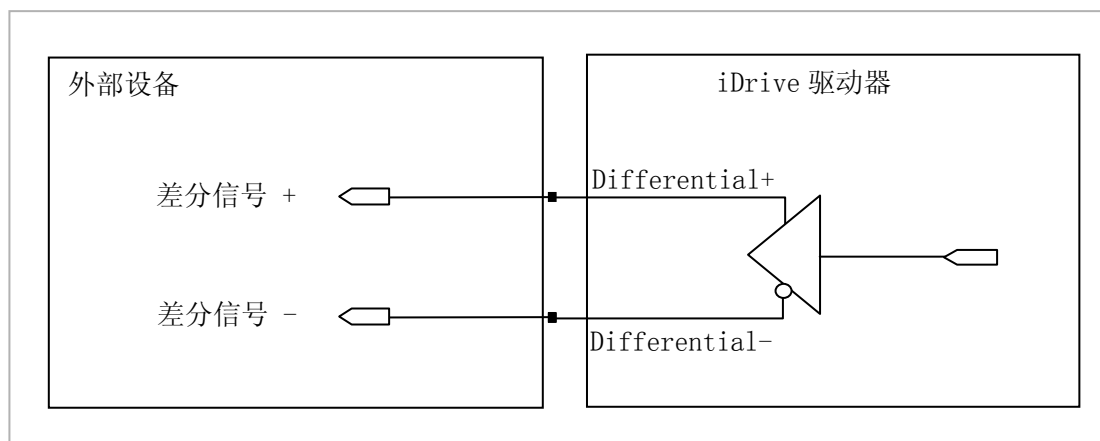


图 23. 差分输出接线方式

光电隔离输入连接方法如下图：

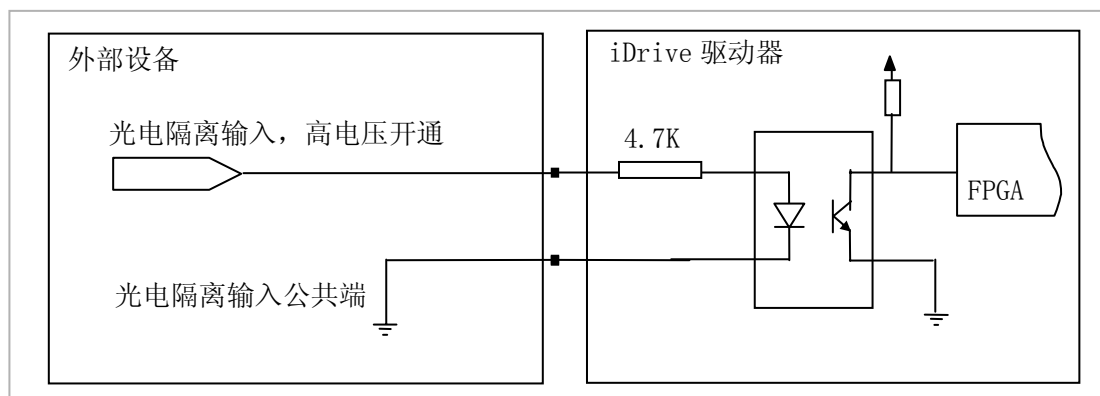


图 24. 光电隔离输入连接方式

光电隔离输出信号连接方法如图 25:

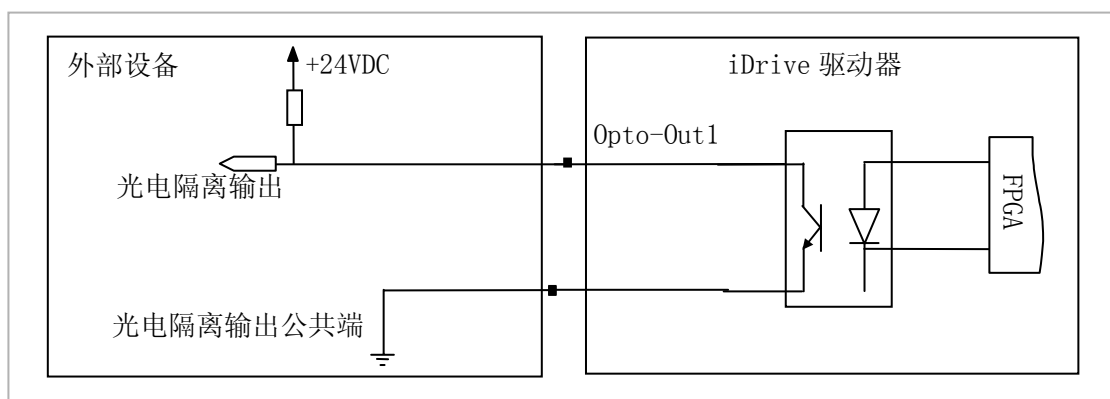


图 25. 光电隔离输出连接方式

iDrive 提供了共 2 路模拟输入接口，模拟输入信号的范围为+10V 到-10V，电路连接方式如图 26 所示。

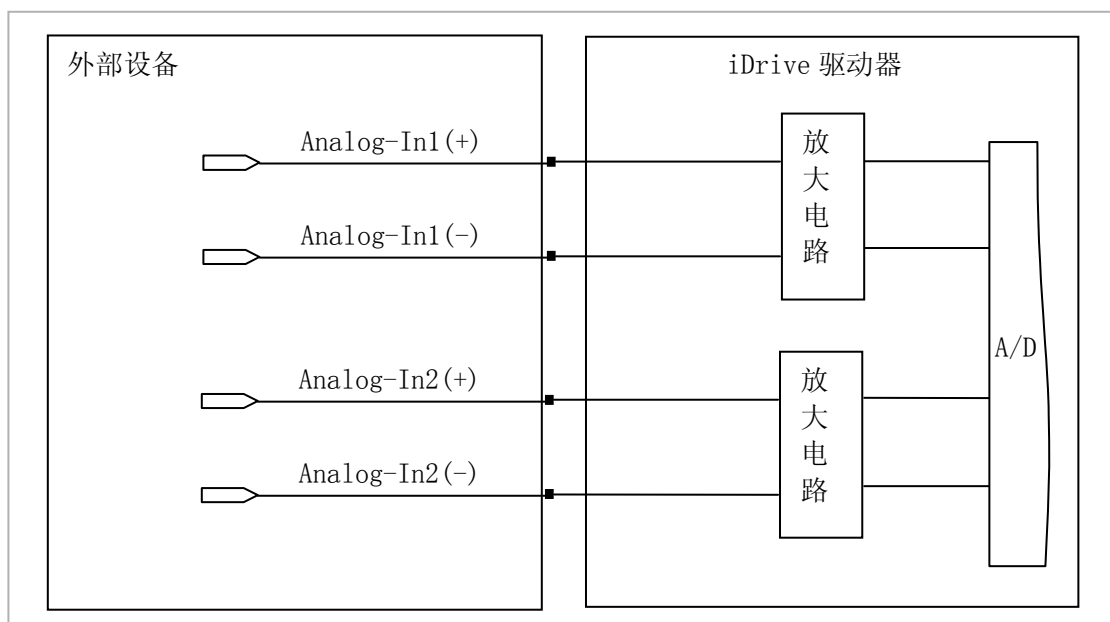


图 26. 模拟输入信号连接方式

3.2.3 地址和通讯设置

J3 号接口是两个拨码开关，分别用于设置驱动器的地址和通讯方式，如图 27 所示：



(1) 地址设置



(2) 通讯方式设置

图 27. iDrive 地址和通讯方式设置

当地址拨码开关旋转到一定的数字，驱动器即设置为相应的地址，请在断电的情况下设置驱动器的地址。

通讯拨码开关的设置如下：

1 ON, 2 OFF: RS232 通讯方式；

1 OFF, 2 ON: RS485 通讯方式。

同样，也请在断电的情况下设置驱动器的通讯方式。

3.2.4 RS232/RS485 通讯接口

J4 和 J5 号接口用于驱动器和上位机的通讯，iDrive 提供了 RS232/RS485 两种通讯方式。其中 RS232 支持单点通讯，即一个 RS232 通讯接口连接一台驱动器。RS485 支持多点通讯，一个 RS485 通讯接口可以联网 31 台 iDrive 驱动器。图 28 显示了驱动器通讯接口和上位机的连接方式：

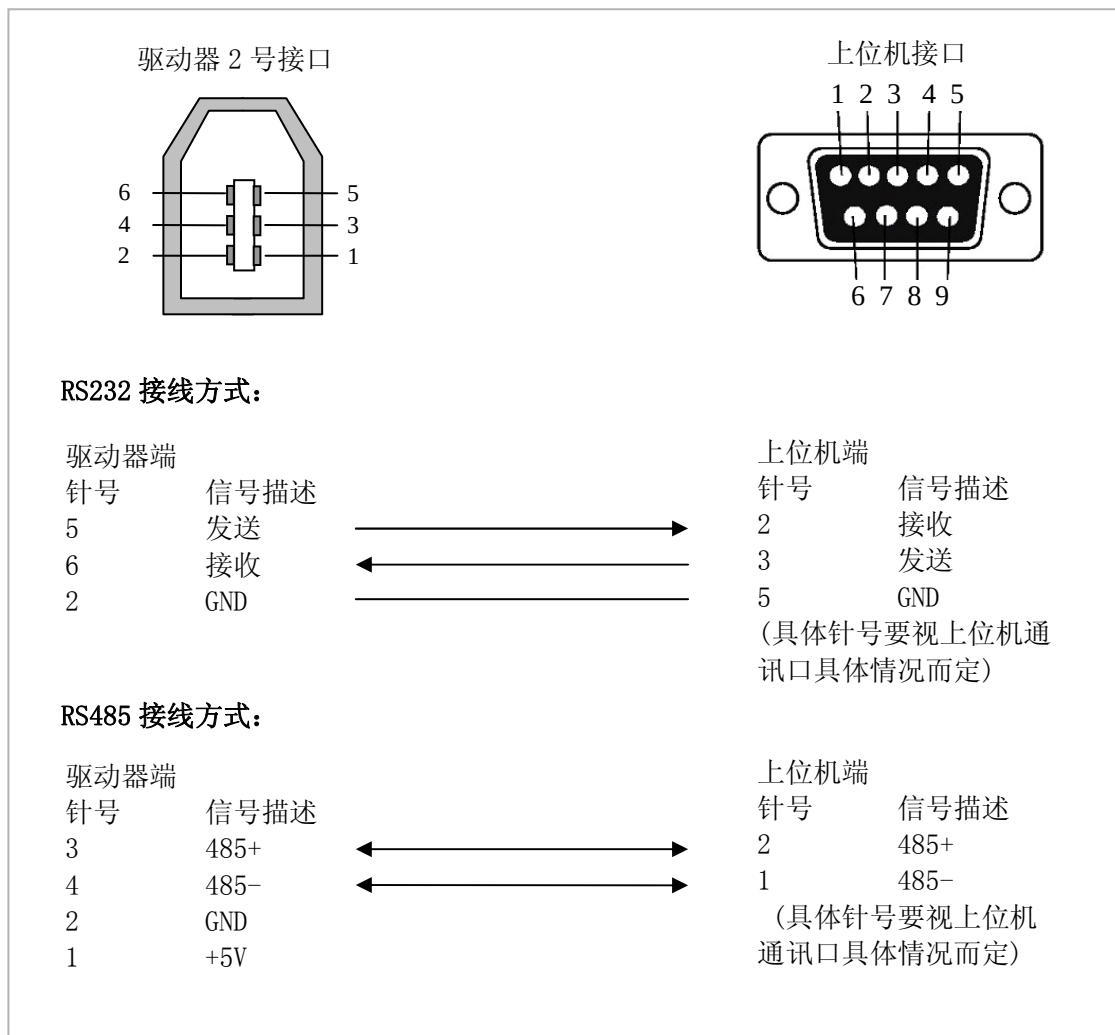


图 28. 驱动器通讯接口的连接方式

RS232 支持单机通讯，一个串口只能连一个驱动器。如果上位机有多个串口，也可以连接多个驱动器，如下图所示。

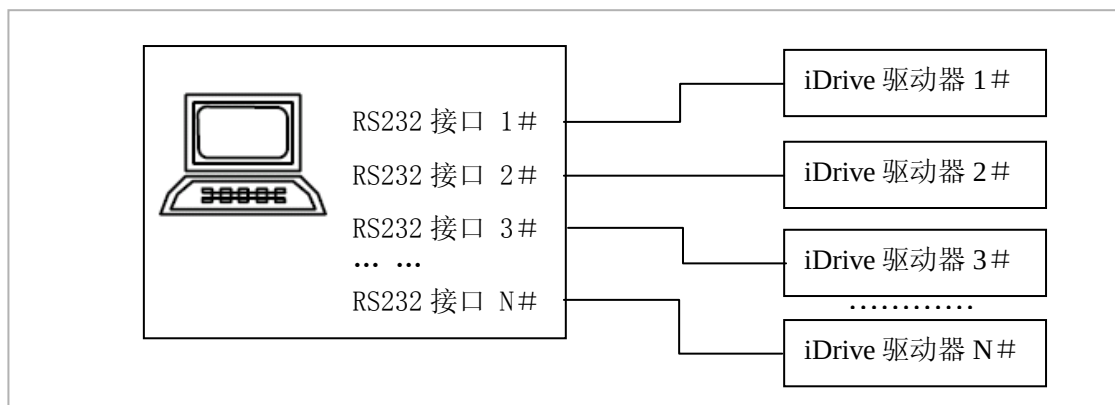


图 29. 上位机用 RS232 口连接到驱动器

RS485 可以实现多达 31 台驱动器的连接，连接方法如下图所示：

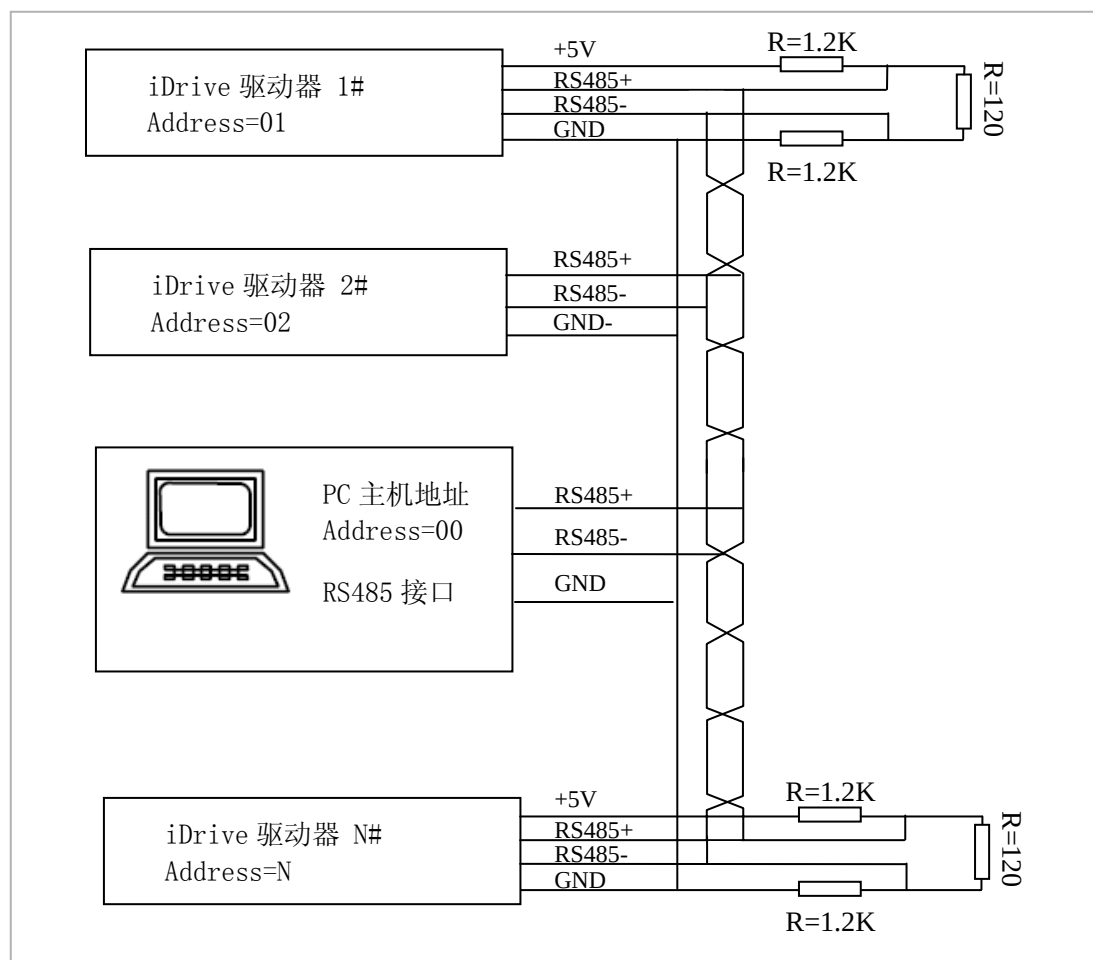


图 30. RS485 网络连接方式

3.2.5 电源接口

J10 接口是驱动器电源接口，包括逻辑低压和驱动高压供电，接口示意图如图 31 所示，其接线方式见表 9。

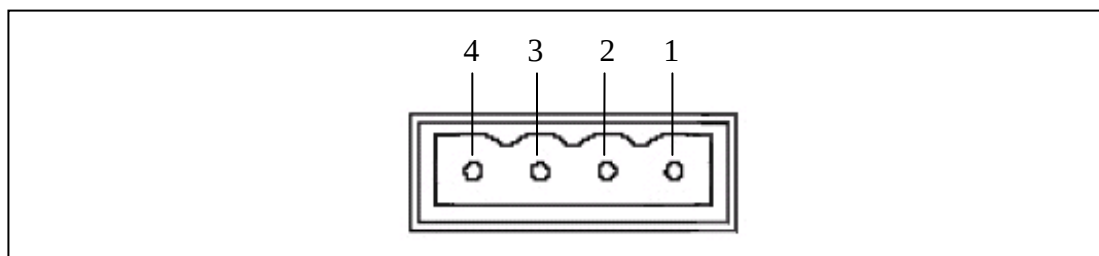


图 31. 电源接口示意图

表 9. 电源接口接线方式

针号	功能说明	备注	说明
1	逻辑低压-	电压范围：12VDC~24VDC 用于逻辑电路的电源	
2	逻辑低压+		
3	驱动高压-	电压范围：24VDC~100VDC 用于电机驱动电路的电源	
4	驱动高压+		

3.2.5 电机制动器接口

iDrive 驱动器提供了电机制动器接口，制动器接口主要用于连接外部的电机制动器，iDrive 输出/输出接口上的制动器接口提供了 250mA 的最大制动器电流。制动器接口采用了集电极开路，除了用于电机制动器的连接之外。此接口也可以用于外部阻抗或者是感抗负载的连接。制动器接口的动作可以通过 motionStarter 上位机调试软件或者是 motionLib 函数库来控制。

制动器接口的连接方式如图 32:

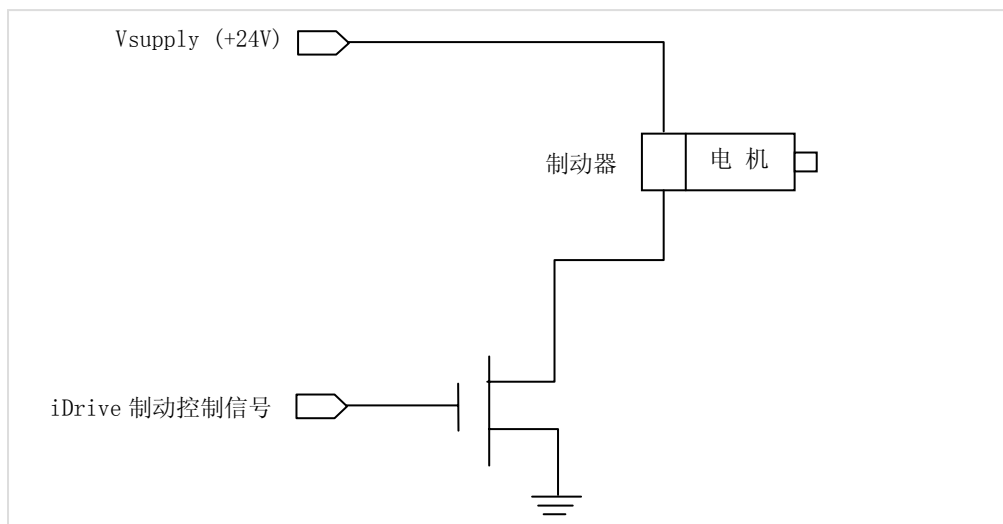


图 32. iDrive 制动器接口连接示意图

3.2.6 编码器接口

J6 和 J8 号接口分别是第 1 轴和第 2 轴的编码器接口(单轴模块只有 J6)，接口示意图如图 33，信号的接口说明如表 10。

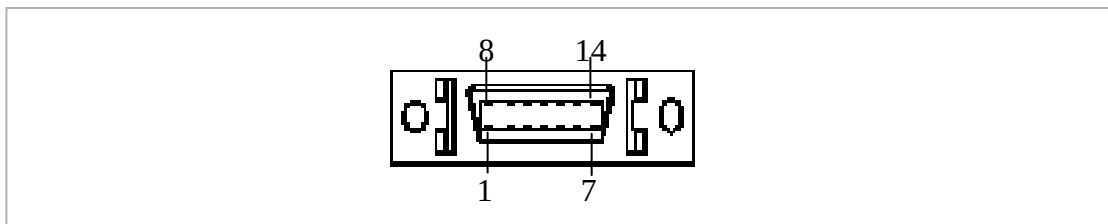


图 33. 编码器接口示意图 (1)

表10. 编码器接口说明

针号	名称	意义
1	VCC	5V 电源输出
2	HA -	霍尔传感器 HA-
3	HA +	霍尔传感器HA+
4	HB +	霍尔传感器HB+
5	HC +	霍尔传感器HC+
6	ENC-A(+)	编码器 A+
7	ENC-A(-)	编码器 A-
8	ENC-B(+)	编码器 B+
9	ENC-B(-)	编码器 B-
10	ENC-Z(+)	编码器 Z+
11	ENC-Z(-)	编码器 Z-
12	GND	信号地
13	HB -	霍尔传感器HB-
14	HC -	霍尔传感器HC-

说明：

- 如果电机编码器不是差分信号，请将编码器各信号线接到相应的编码器接口正端，并且将驱动器上编码器接口的负端悬空。

下图说明了编码器接口焊盘编号的定义，将编码器接口插到驱动器上。从上往下看编码器接口的焊盘编号如下图所示。

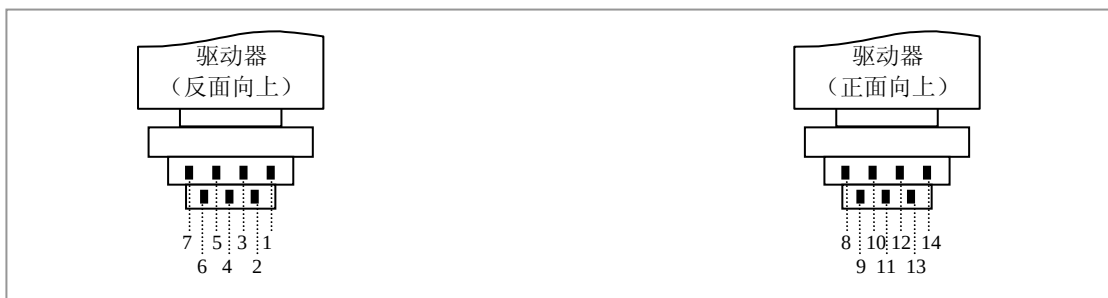


图 34. 编码器接口焊盘定义示意图 (2)

3.2.7 电机动力线接口

J7 和 J9 接口是第 1 轴和第 2 轴电机的动力线接口(单轴模块只有 J6)，接口示意图如图 35，信号的接口说明表 11。

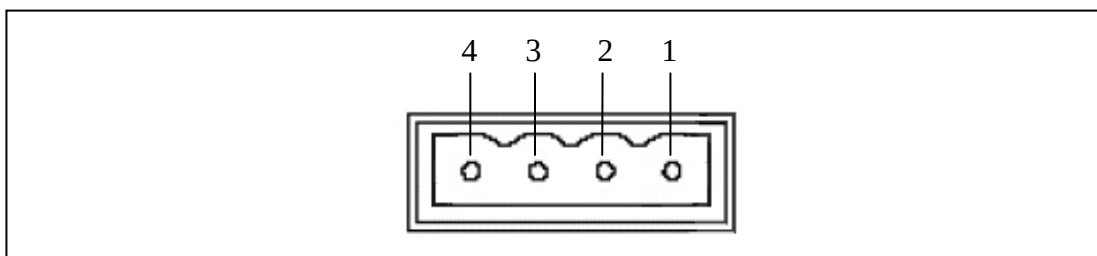


图 35. 电机动力线接口示意图

表11. 电机动力线接口说明

针号	功能	备注
1	接地	
2	电机 U 相	
3	电机 V 相	
4	电机 W 相	

3.3 第一次上电

为了设置驱动器，首先必须与驱动器进行通讯，最简单的方法是在PC 和驱动器之间通过RS232 串口进行联机。在第一次上电前，请检查以下事项：

- 供电电源连接及电压值
- 电机编码器信号和动力电缆连接
- 通讯电缆连接
- RS232/RS485 通讯方式设置是否正确设置为 RS232
- 在 PC 上安装 montionStarter 调试软件，通过串口连接驱动器（见驱动器调试步骤）

驱动器连接完成后就可以开始系统的调试。

	注意：在连接驱动器的过程中，请注意人体静电，如果系统及人体带有静电并引起放电，有可能损坏驱动器
---	---

	警告：在连接驱动器之前，请确认电源线连接正确，否则会损坏驱动器。特别需要注意的是，驱动器不能带电插拔。否则有可能损坏驱动器。
---	--

4. iDrive 的调试

4.1 motionStarter 调试软件的安装

在驱动器的硬件连线安装完毕之后，下一步可以进行驱动器的调试。在驱动器的调试之前，必须先在上位机安装调试软件 motionStrater。

montionStrater 是一款基于 PC 的用于 iDrive 系统调试的软件。电机的控制参数、输入/输出口的设定以及独立运行模式运动程序的编程都可以用它来完成。当系统调试完毕之后，所有的控制参数都可以备份到硬盘上，以供以后恢复使用。参数文件以 .MAP 为扩展名。

在随机附带的光盘上找到 motionStarter 的 Setup.exe 文件，双击开始安装。安装完毕后可以在 START->PROGRAM->NORTINOTECH 找到 montionStarter。单击打开 montionStarter，系统会显示 montionStarter 的主界面，如图 36 所示：



图 36. montionStarter 主界面

4.2 iDrive 驱动器控制回路介绍

为了更好地进行驱动器控制参数的调节，在使用 motionStrater 进行参数调节之前，我们先来了解 iDrive 控制回路的组成情况。iDrive 采用了 PID 加前馈的控制模式，控制回路如图 37 所示。控制回路的外环为位置环，其次为速度环，内环为电流环。通过控制位置和速度两个状态变量，位置回路能提供较好的性能。加上速度前馈和加速度前馈控制功能，加快大惯量系统的动态响应性能。由于编码器不能直接输出速度信号，电机的运行速度是通过速度观测器来获得的。



警告：在进行系统控制参数的调试和优化之前，调试人员至少需要了解并且熟悉 PID 控制回路中各个控制参数的作用和控制原理。没有控制理论基础或经验的人员进行系统调试有毁坏驱动器的危险。

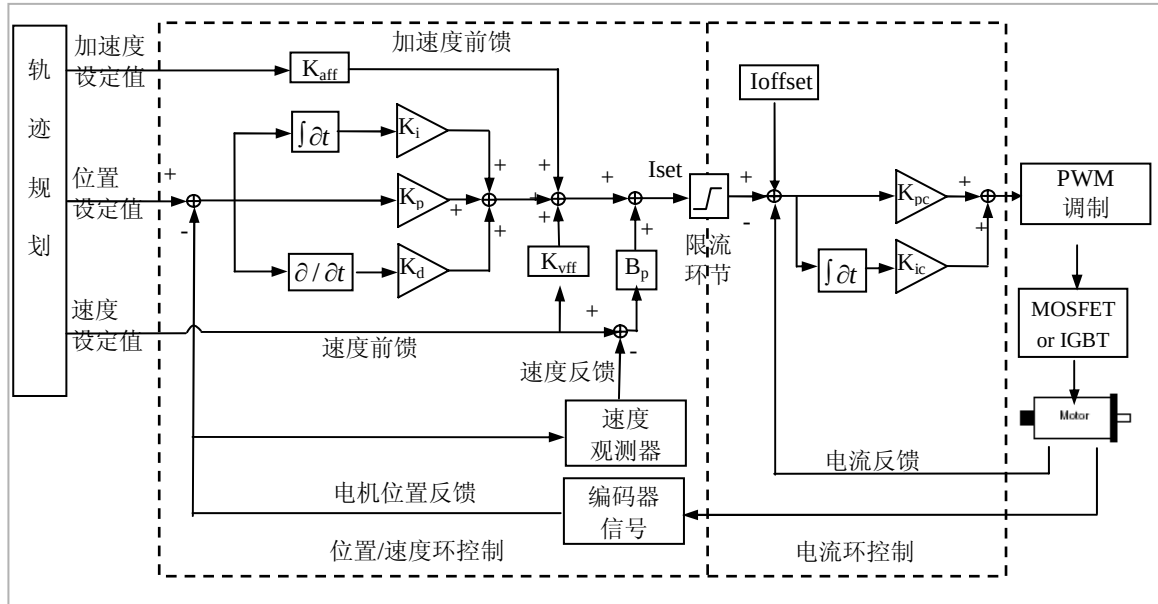


图 37. iDrive 驱动器控制回路

电机的电流及电压按以下方程计算得到：

$$I_{set} = I_{offset} + K_p \cdot P_{error} + K_i \cdot \sum P_{error} + K_d \cdot \Delta P_{error} + B_p \cdot V_{error} + J_p \cdot A_{error} + K_{vff} \cdot V_{set} + K_{aff} \cdot A_{set}$$

$$V_{motor} = K_{pc} \cdot I_{error} + K_{ic} \cdot \sum I_{error}$$

控制环回路和控制方程中各变量的意义如下表所示：

表12：驱动器控制回路变量说明

变量名	意义
I_{set}	电机电流设定值
I_{offset}	电流偏置，通常用于补偿垂直轴重力的影响
K_p	位置比例增益
K_i	位置积分增益
K_d	位置微分增益
B_p	速度比例增益
K_{vff}	速度前馈增益
K_{aff}	加速度前馈增益
P_{error}	位置误差 = $P_{set} - P_{actual}$
$\sum P_{error}$	位置误差之和
ΔP_{error}	位置误差增量

V_{error}	速度误差 = $V_{set} - V_{actual}$
A_{error}	加速度误差 = $A_{set} - A_{actual}$
P_{set}	位置设定值
V_{set}	速度设定值
A_{set}	加速度设定值
P_{actual}	位置真实值
V_{actual}	速度真实值
A_{actual}	加速度真实值
V_{motor}	电机线圈电压
K_{pc}	电流环比例增益（在motionStrater中为Kp，这里为了区分位置环增益，故写为Kpc）
K_{ic}	电流环积分增益（在motionStrater中为Ki，这里为了区分位置环增益，故写为Kic）
I_{error}	电流误差 = $I_{set} - I_{actual}$

在系统调试之前，需要通过 motionStrater 设置系统参数，系统参数设置的详细情况请参见下一章节。需要设置的主要参数如下所示：

- 电机类型设置；
- 每转编码器读数和编码器反馈类型；
- 最大速度误差和最大加速度误差（主要用于监测在电机运动过程中比较位置和速度的设定值和实际值之间的误差。如果误差超过设定的误差限系统将会报警）；
- 电流限制的平均值和峰值以及平均值的限制时间设置；

以上这些参数设置完成以后就可以进行电流环、速度环和位置环的调试了。图 38 显示了经过简化的控制器示意图，图中给出了需要调试的关键控制参数。

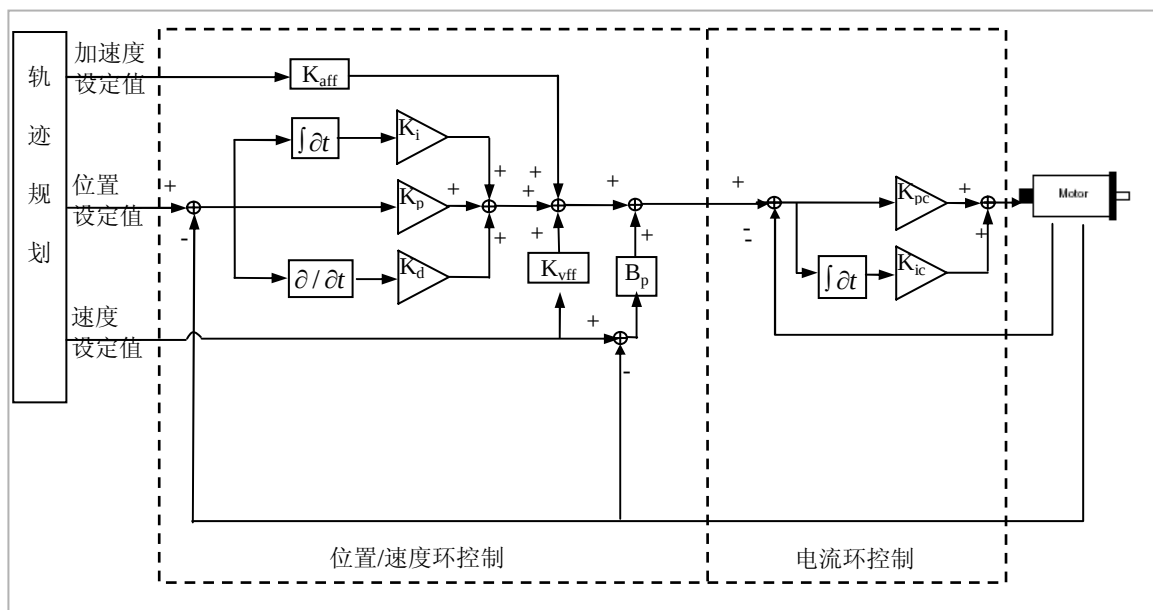


图 38. iDrive 驱动器控制回路

需要调试的控制回路主要控制参数如下：

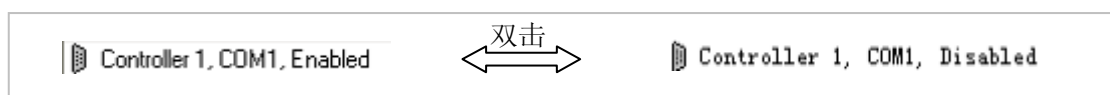
Kp：	位置比例增益
Ki：	位置积分增益
Kd：	位置微分增益
Bp：	速度比例增益
Kvff：	速度前馈增益
Kaff：	加速度前馈增益
Kpc：	电流环比例增益
Kic：	电流环积分增益

4.3 用 motionStarter 调试 iDrive

4.3.1 连接驱动器和上位机

确认驱动器和上位机连接正确无误之后，可以点击主界面(图 36)上的连接  按钮打开连接设置对话框，如图 37 所示。如果是第一次连接，请单击“增加连接”按钮打开如图 39 的连接属性设置对话框。确定连接属性及通讯口设置正确无误，点击“确定”返回到连接设置对话框。在驱动器已经正常上电并正确连接通讯线缆的情况下点击“连接驱动器”对话框的“确定”按钮便可以连接驱动器。如果有多个驱动器同时存在，上位机会连接到多个驱动器。

需要说明的一点是如图 37 的连接驱动器对话框，只有当驱动器状态处于“Enabled”的状态下才能成功连接驱动器。而只要双击驱动器名称一栏，驱动器的状态可以在 Enabled 和 Disable 之间切换，如图 38 所示。



驱动器名称是非常重要的，它是表示驱动器地址的重要标识。特别是在 RS485 多点通讯的情况下，当使用 RS232 通讯时，地址会被上位机忽略而用通讯口来标识不同的驱动器。



注意：不管是用 motionStrater 调试界面还是 motionLib 函数库，使用 RS232 通讯时驱动器和上位机界面都会自动地将通讯速率调整到最大即 115kbps，即使是驱动器启动的时候通讯速率只有 9600bps。iDrive 驱动器和上位机软件支持通讯波特率的自动调整。同样在使用 RS485 通讯的时候，驱动器和上位机的通讯会自动调整到最大值 921.6kbps。



图 37. 连接设置

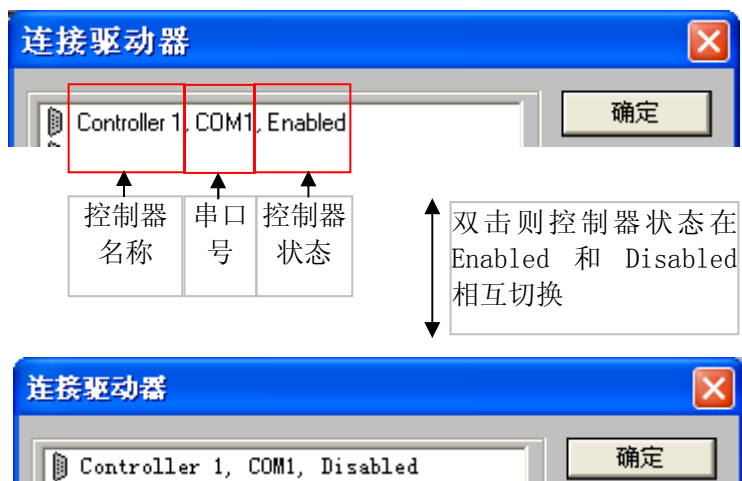


图 38. 驱动器状态切换



图 39. 添加连接

4.3.2 motionStrater 主界面

通过上一节的操作将驱动器连接到上位机界面以后，上位机界面将显示已经连接上的驱动器，如图 40 所示。同时系统显示如图 41 的系统调试主界面。系统调试主界面的主要内容有：

 系统设定	系统及电机参数设定 包括驱动器操作模式、电机类型、系统高低限位、误差限制、电流限制以及输入/输出口的调试及状态显示等，同时显示驱动器的版本信息
 电流环	电流环调试 利用给定的电流环设定值调试电流环控制性能及其控制增益
 速度/位置环	位置环和速度环调试 位置环和速度环控制性能及其控制增益及简单运动的调试
 运动设计	运动程序编程及调试 运动程序编程及将编译好的运动程序下载到驱动器，以供独立运行模式应用。同时还可以进行 iDrive 的两轴运动调试
 故障诊断	故障诊断 用于判断驱动器的工作状态以及快速故障诊断
 轴间参数复制	系统参数拷贝 将系统参数从一个轴拷贝到另一个轴，包括轴设置参数、电流环、速度环及位置环的控制增益值等
 更新参数	更新系统参数 将 DSP 中的系统参数保存到 Flash ROM 中
 下载参数	下载系统参数 将系统参数从本地硬盘 .MAP 文件中下载到驱动器 Flash ROM 中
 保存参数	保存系统参数 将驱动器的系统参数保存到本地硬盘，生成 .MAP 文件

 下载程序	下载系统程序 下载系统 DSP 程序和 FPGA 配置文件到驱动器，用于更新系统程序及 FPGA 配置文件。还可以用来下载运动控制程序到驱动器。
--	--

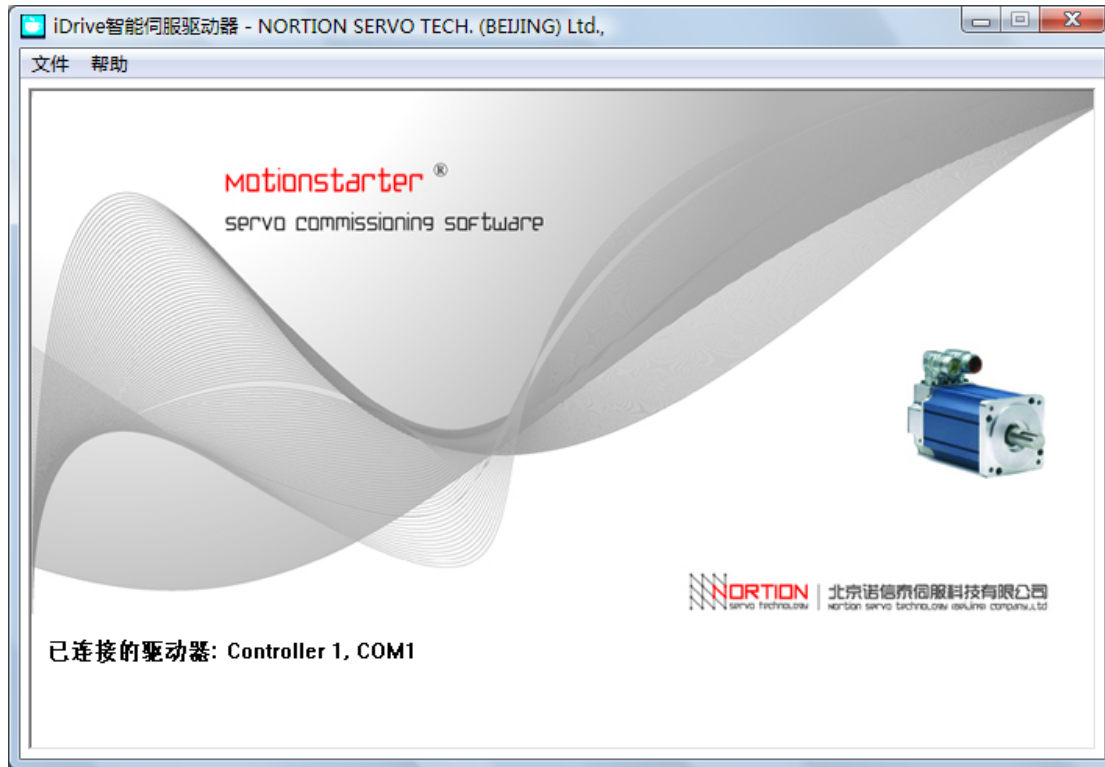


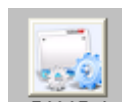
图 40. 显示已连接上的驱动器



图 41. 系统调试主界面

	警告：在看完本手册之前，切勿对驱动器进行操作，不合规的操作可能会损坏驱动器，带来不必要的损失。
---	---

4.3.3 系统及电机参数设置



系统设定

点击系统调试主界面(图 41)的系统设置按钮，打开如图 42 的系统和电机参数设置窗口。在系统及电机参数设置窗口中，通过选择不同控制器和轴可以显示不同控制器和轴的参数。通过系统和电机参数设置窗口，可以了解和设置以下参数：

图 42. 系统及电机参数设置

显示驱动器的信息：

- 驱动器高压供电电压
- 驱动器产品信息和序列号
- 驱动器软件版本信息
- 输入端口的状态，包括光电隔离输入、差分输入、模拟输入（输入/输出口的接线请参见 3.2 章节）

需要设置的驱动器信息：

- 操作模式
 - 网络指令模式（RS232 或 RS485），用于上位机发送运动命令的控制模式
 - 独立运行模式，驱动器脱离上位机运行，电机的运动有存储于 Flash ROM 中的运动程序控制
 - 脉冲/方向模式，

- 模拟信号方式
- 控制模式
 - 位置闭环模式
 - 速度闭环模式
- 电机类型选择
 - 交流伺服电机(带有霍尔信号，直流无刷电机也可以用此选项)
 - 交流伺服电机（没有霍尔信号）
 - 直流有刷电机
- 电机编码器的设置和编码器状态

用于设置编码器是多少个计数每转，iDrive 驱动器采用编码器 4 倍频的技术，如果编码器是 2500 线/转，则将编码器设置为 10000 计数/转。编码器状态栏显示了编码器的读数和霍尔传感器的状态(对于带有霍尔的伺服电机而言)。如果不知道每转编码器的读数是多少，可以手动将电机转子旋转一转，通过观察编码器的读数得到每转编码器的读数。

电子齿轮是用于脉冲/方向控制模式，它表示了输入脉冲和电机编码器之间的比例关系。

逆反馈的设置使得编码器的方向与霍尔传感器和电机三相的方向相匹配。大多数情况下，当电机旋转且编码器读数正向增加时，霍尔传感器的变化序列为 100->110->010->011->001->101->100，则不用选中逆反馈选项。如果霍尔传感器的变化序列相同时，电机编码器反向增加，则需要选中逆反馈选项。

图 43 显示了直流无刷电机的三相六步控制模式，其中显示了霍尔传感器的信号和电机每相之间的关系。

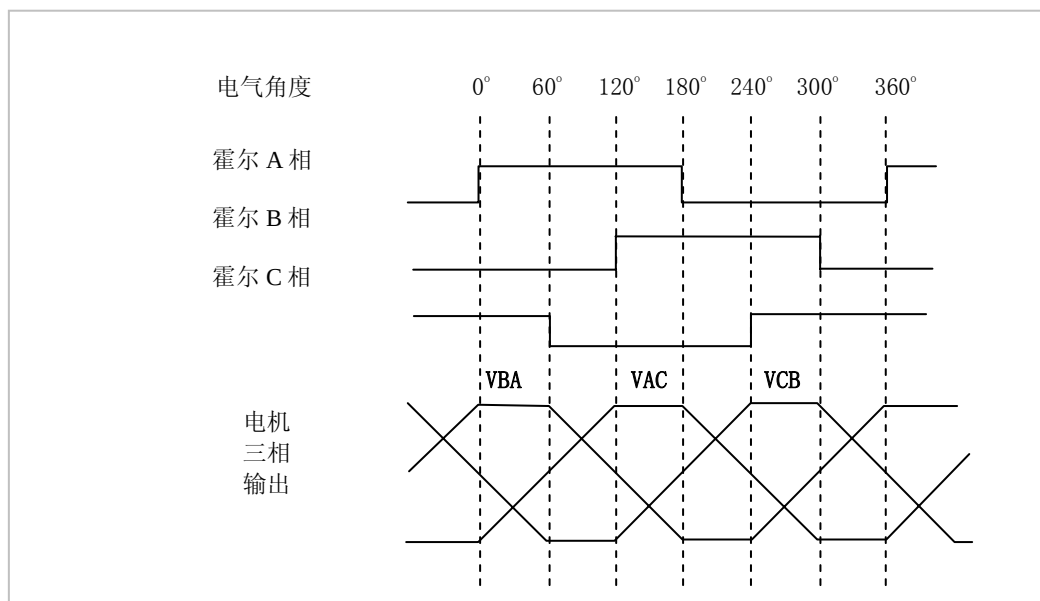


图 43. 直流无刷电机 3 相 6 步控制模式

- 误差限设置

误差限设置包括最大位置误差和最大速度误差。主要用于监控电机按照一定的轨迹运行的时候的最大误差限。如果速度或位置误差超过最大误差限制，则系统会报警且取消电机的使能状态以保证加工精度和机械系统的安全。

■ 电流限制

为了保护系统过流，电流限制选项设置了峰值电流限制、平均电流限制和平均最大电流持续时间的设置。设置这些参数用于避免损坏驱动器或电机。

■ 无霍尔控制

无霍尔控制选项包括霍尔追踪时间和电机电极对数的设置，主要用于 iDrive 驱动器用于控制不带霍尔的伺服电机使用，如图 44 所示。霍尔追踪时间的设置成默认值（3 秒）就可以，而电机的电极对数必须保证正确无误才能保证驱动器能正常驱动电机运转。如果用户不知道电机电极对数的参数，请参考电机厂家所提供的参数。另外，在使用直流有刷电机或交流伺服电机时，每转编码器的读数设置必须保证正确才能确保系统能正常运转。

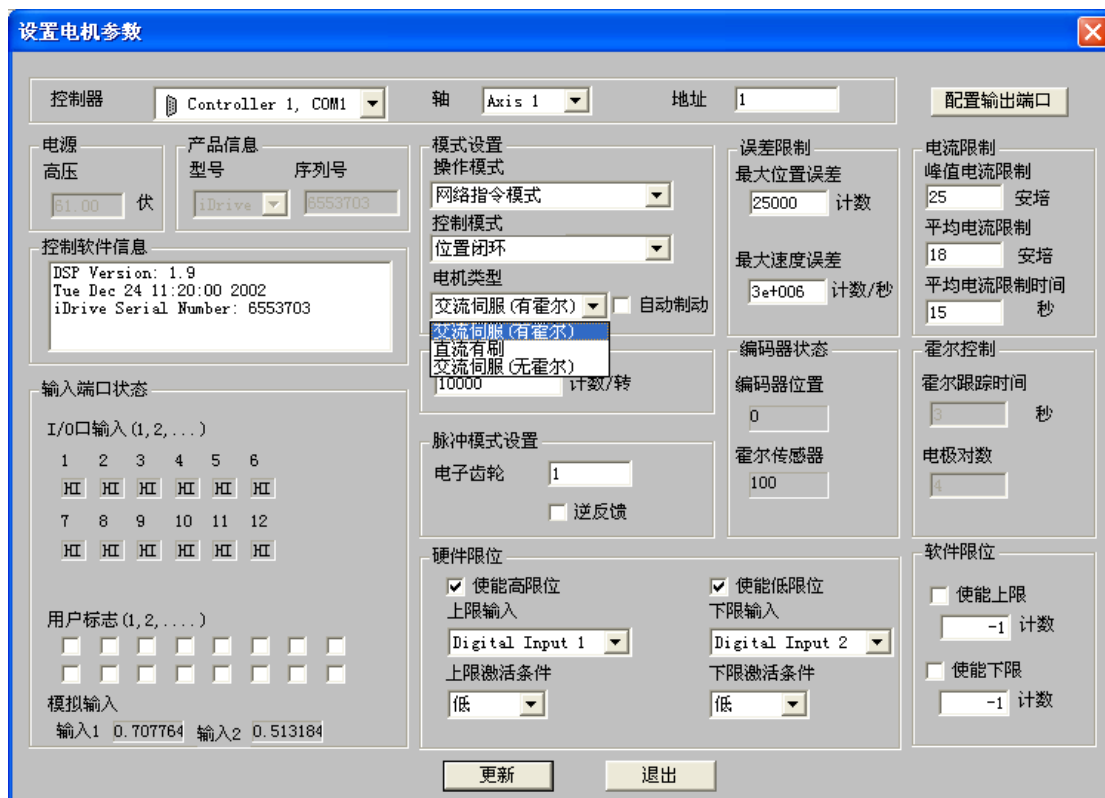


图 44. 伺服电机参数设置

■ 硬件限位

硬件限位包括高限位和低限位，所需要设置的参数有高限位使能、低限位使能、高低限位输入是通过哪个 I/O 口实现、以及 I/O 的高低限位是高电平有效还是低电平有效。高低限位通常用于保护系统机械装置免受冲击损坏。用于高低限位的输入端口请参见输入/输出端口的接线章节。

■ 软件限位

软件限位是通过编码器的读数来限制电机的运行不能超过高低位置的限制，为机械系统提供了最初一级的保护。

■ 检查电机接线

当电机接线完毕之后，可以将电机各接口连接到相对应的驱动器接口上。接通驱动器供电，这时驱动器进入工作状态。通过手动转动电机的转子，同时观察电机编码器读数和霍尔传感器的变化来检查电机连接是否正确。

检查编码器：手动转动电机一转，观察电机的编码器读数是否是转动一转的读数，如果是则说明编码器接线正确，否则需要检查编码器的接线。

检查霍尔传感器：转动电机，观察霍尔传感器信号的变化应该是以下两种序列之一。

- (1) 100→110→010→011→001→101→100
- (2) 100→101→001→011→010→110→100

如果霍尔传感器的信号同时出现了 000 或 111 的信号，则说明霍尔传感器的连接有误，需要检查接线。

另外，在转动电机时同时观察编码器和霍尔信号的变化，如果编码器正向增加时霍尔信号的变化序列时以上的(1)，则需要将编码器的逆返回选项选中，相反则不用。



警告：如果霍尔信号或者是电机相序接错，极有可能导致系统飞车。所以在电机连接机械系统之前，请确认电机接线和工作都正常之后再将电机连接到机械系统。

当系统参数设置完毕之后，可以将系统参数更新到驱动器。“更新”按钮就是用于将设置的系统参数更新到驱动器。

点击电机输出设置按钮“配置输出端口”可以进入输出端口的设置。



注意：在驱动器更新参数的时候，请不要切断电源或断开通讯。否则系统参数有可能被损坏而导致驱动器不能正常运行。

4.3.4 输出端口设置

输出口状态设置用于设置驱动器各个输出口的状态，包括使能、周期还是非周期性。如果是周期的输出，还可以设置高电平和低电平的时间。输出端口状态设置请参见图 45。



图 45. 输出端口状态设置

4.3.4 电流环调试

如图 46 所示，电流环是一个 PI 控制器，需要调试的参数有两个。分别是电流环比例增益和积分增益。

这两个控制参数的主要作用是：

电流环比例增益：

其作用是加强电流环控制作用，加快控制回路的响应速度，减少电流环的控制误差。电流环比例增益越大，控制作用越强，响应速度越快，电流环控制误差越小。但是增大电流环比例增益不能消除误差到零；

电流环积分增益：

其作用是加强电流环的控制作用，消除电流环的控制误差。电流环积分增益越大，积分控制作用越强，电流控制误差越小。

	警告：实施电流环调节的时候，系统处于速度和位置开环的状态。改变电流环的控制参数可能引起系统的不稳定。在调试电流环之前要确认电流环的调试不会引起机械故障或者是有潜在的风险，如果有系统不稳定，请做好系统紧急停车的准备。
---	---

单击主界面的电流环调节按钮进入电流环调节界面，如图 46 所示。通过这个界面用户可以对驱动器各轴的电流环进行调节，并可以实时显示电流环的调节结果以评估电流环的性能。

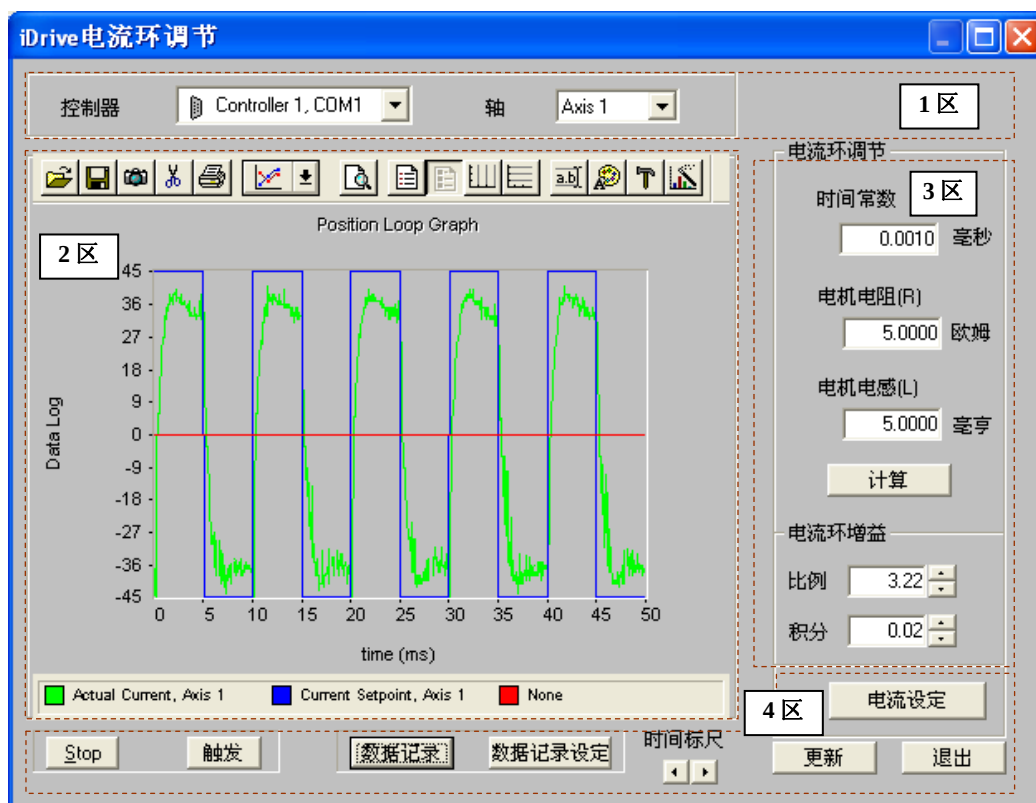


图 46. iDrive 电流环调节界面

下面先介绍一下电流环调节界面的功能划分，如图 46 所示。

- 1 区：控制器和轴选择区，用于选择想要调试的目标轴；
- 2 区：电流曲线实时显示区，用于实时显示电流环的设定值和实际值曲线；
- 3 区：控制增益区，显示并可增加或减少控制增益的大小；
- 4 区：控制区，用于控制整个参数调节过程。

各按钮的功能如下：

电流设定: 设置电流环给定值

Go/Stop: 启动/停止电流环开始工作

触发: 触发电流环数据记录

数据记录: 触发在 2 区显示电流曲线

数据记录设定: 设置电流环数据记录

了解了界面的组成，我们可以开始电流环的调试工作。电流环的调节步骤如下：

第一步：检查系统连接及参数设定

在调节电流环之前，必须保证驱动器与电机、电源的连接正确且电源已经接通；必须保证电机和编码器参数已正确设定。驱动器接口的连接和设定的相关细节可参见之前的相关章节。

第二步：设定电流环控制增益初始值

- 比例增益 Kpc
- 积分增益 Kic

由于不同的电机会有不同的参数设定，在开始阶段可以将比例增益设置到一个较小值，比如0.5，将积分增益设为0。

在调节控制增益的时候，往往是先将积分增益设为0。先调节比例增益慢慢变大，当电流环的响应达到可以接受的程度再来增大积分增益，减少控制误差。当然，要是系统响应达到最优，整个过程需要反复进行多次。



注意：在初次进行电流环调节时，应尽量将电流环的控制增益设定在较小的值，然后再根据电流环的调节效果调节控制增益的大小。如果控制增益太大会影响系统的稳定性。

第三步：电流给定值设定

在图46所示的电流环调节对话框中，单击电流设定按钮，弹出电流控制对话框（如图47）。用户可以通过电流设定对话框设定以下参数：

- 1) 控制器：选择要调试的控制器；
- 2) 轴：选定要调试的轴；
- 3) 电流设定：电流设定值，建议低于1安培，典型值如0.3安培；
- 4) 周期设定：典型值为10毫秒，同时应选定周期性（选择周期性则电流设定值为周期性方波信号，否则就为阶跃信号）。

上述设置完成后，点击更新按钮以保存设置，然后点击退出按钮返回到电流环调节对话框



图 47. 设置电流环给定值

第四步 数据记录设定

在电流环调节对话框中（图46），单击数据记录设定按钮，弹出记录数据对话框（如图48）。在此对话框中，用户可以设定与记录数据有关的参数：

- 1) 选择控制器、控制轴；
- 2) 将选定轴的通道A、通道B分别设为真实电流、设定电流；也可以通过快速设定栏目来设定记录电流值的通道；
- 3) 设置采样时间为0.05毫秒。0.05毫秒是系统最快的采样周期(20kHz)，因为电流环调节的响应速度较快，因此应选取最快的采样周期；当然也可以设定为更长的采样时间。每个通道能记录1024个数据；
- 4) 选中使能数据记录；
- 5) 在自动触发一栏，若轴被选中，电机启动时，该轴的运动数据将被自动存储并传递到上位机实时显示；

上述设置完毕后，点击对话框中的更新按钮以保存设置，再点击退出按钮返回到图6所示的电流环调节对话框；



图 48. 设置电流环数据采集通道

第五步 启动电机, 观测电流环调节效果

以上参数设置完成之后，可以启动电流环的调节。

点击电流环调节对话框中的启动（Go）按钮，驱动器将根据设定好的参数进行电流环的调节，同时启动（Go）按钮转变为停止（Stop）按钮。再次点击此按钮，调试停止，而此按钮又转变为启动（Go）按钮。这样，此按钮可以在启动、停止之间不断切换来控制电流环的调节过程。



注意：启动电机后，如果电机运动异常或者不稳定，请立即停止电机运动。将电机控制增益调到更小的值后再试。

第六步 数据载入与观测

点击电流环调节对话框中的数据记录按钮，可以将测试数据以图形方式显示出来。

当点击启动（Go）按钮后，将触发数据记录功能，数据记录将从0开始，最大记录数据个数为1024（相应的记录时间长度=采样周期×1024）。通过调节电流环调节对话框中的时间尺度（Time Base）按钮，可以缩短或增加时间轴长度。另外实时显示的数据也可以保存到本地，以供详细分析之用。

第七步 实时在线调试

- 1) 点击启动（Go）按钮，然后再点击数据记录按钮，则显示当前设置的期望电流与实际电流响应曲线;
- 2) 使用滚动条分别调节比例增益(Kpc)及积分增益(Kic)，则在新参数下的系统响应曲线就会实时地显示出来;
- 3) 不断调节增益参数，直到得到满意的结果为止。最后, 点击储存按钮保存调节后的增益参数。



注意：如果真实电流超出平均电流限制值一定时间（系统参数设定对话框中设有平均电流限制时间），系统会出现平均电流错误而禁用相应的电机。为了避免出现此错误并导致电机被禁用，可以快速地执行点击启动按钮——> 点击停止按钮——> 点击记录按钮操作流程。用这种方法，可以保证不使电机过热，同时仍可得到峰值电流的数据曲线。

第八步 储存所有参数

电流环调节完毕后，将 DSP 中的参数更新到 Flash ROM 中保存。

电流环调节策略：

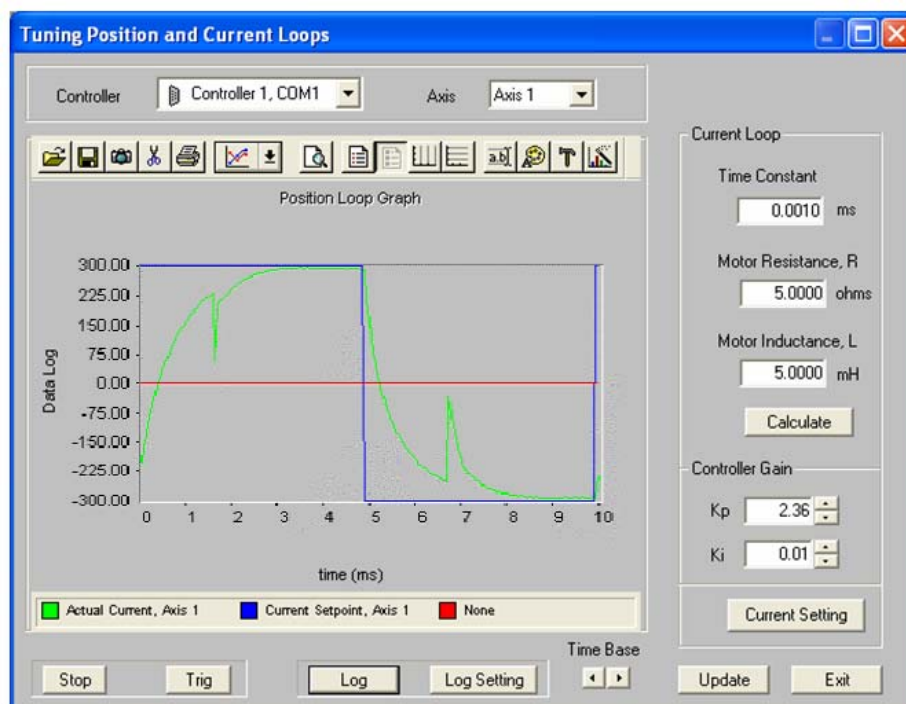
前面已经提到，在电流环的调节过程中，我们首先将电流环的比例增益设置到比较小的值，将积分增益设置为零。开始根据电流曲线实时调试电流环，先调节比例增益使得电流环的响应达到一个可以接受的程度，然后开始调节积分增益。比例增益和积分增益的调节通常是一个反复的过程，需要经过多次调节才能使得电流环的响应达到最优。以下是两个电流环调节的实例。

电流环调节实例（1）

1) 过阻尼情况

图49给出了一个电流环调节实例。从图中的响应曲线可以看到，此时电流真实值远小于电流设定值，说明系统阻尼较大，系统响应较慢。此时，可以同时或分别增加比例增益及积分增益，以加强系统的控制作用、加快系统响应速度。

为了提高系统响应速度，此时应逐步增加系统的比例增益直到比例增益的增大对上升时间变化不很明显为止。注意，过大的比例增益可能使系统产生震荡，因此不能使比例增益过大。但电流环比例增益太大而导致过调时，应恰当减少比例增益和积分增益。另外，应逐步增加积分增益以减少稳态误差，直到系统出现微小超调量为止。



49. 电流环调节结果（过阻尼情况）

2) 欠阻尼情况

图50给出了电流环调节后系统具有较大超调量的结果。与过阻尼的调节方式相反，此时应该适当减小比例增益或积分增益直至超调较小位置。

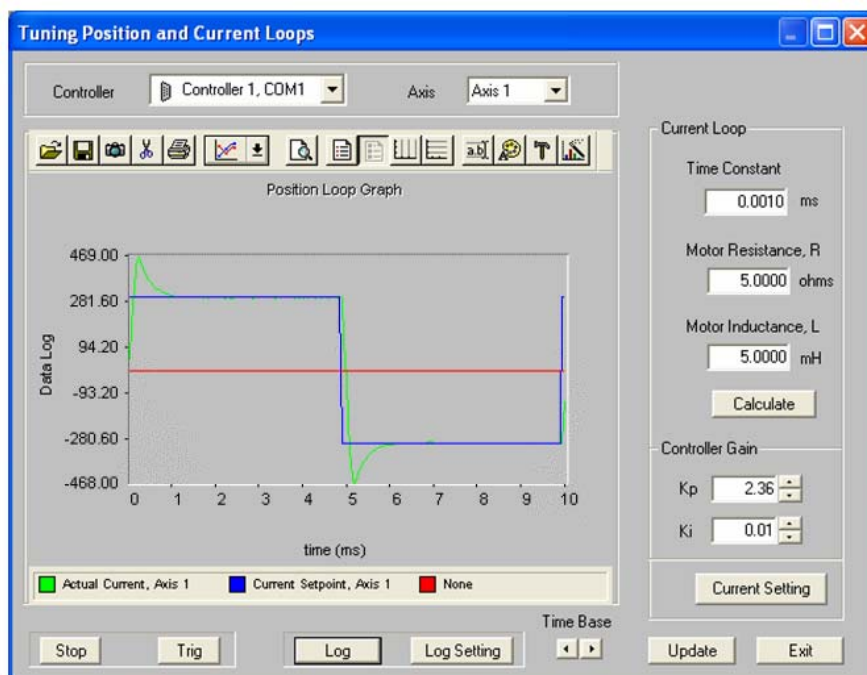


图 50. 电流环调节结果（欠阻尼情况）

3) 较好的电流环调节状态

图51给出了一个调节较好的电流环调节结果。从图中可以看出此时系统的超调量较小，响应较快。

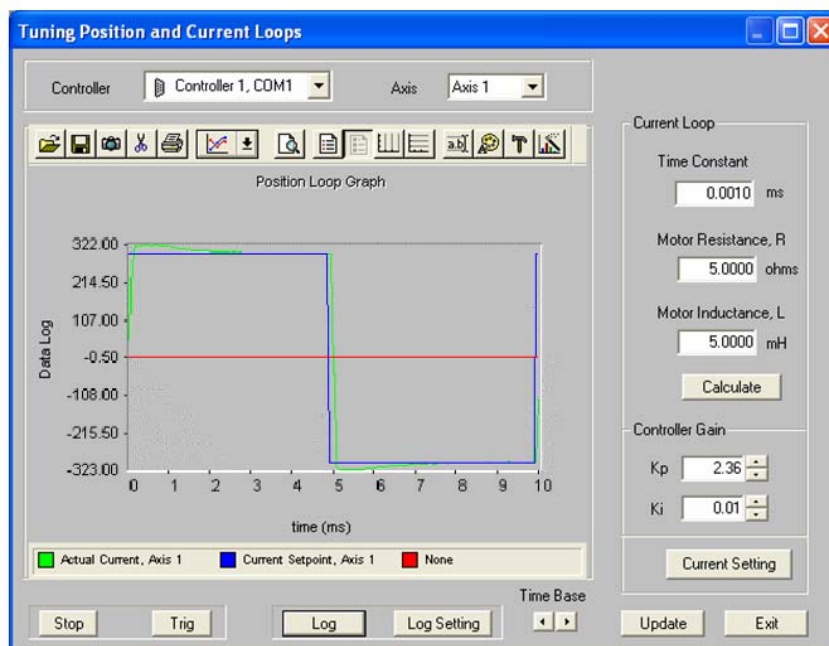


图 51. 电流环调节结果（较好调节情况）

电流环调节实例（2）

需要说明的一点是电流环调节的效果还与电机的性能的好坏有很大的关系，电机的性能越好，电流环的就能得到更好的调节效果。

下面给出了使用另一种电机的情况下得到的电流环调节结果。

1) 选择较小的比例增益和积分增益

如图 52，此时系统比例增益为 0.38、积分增益为 0。由于控制作用较弱，系统的响应比较慢，电流环的误差也比较大。此时的策略是增加比例增益，加强系统的控制作用。

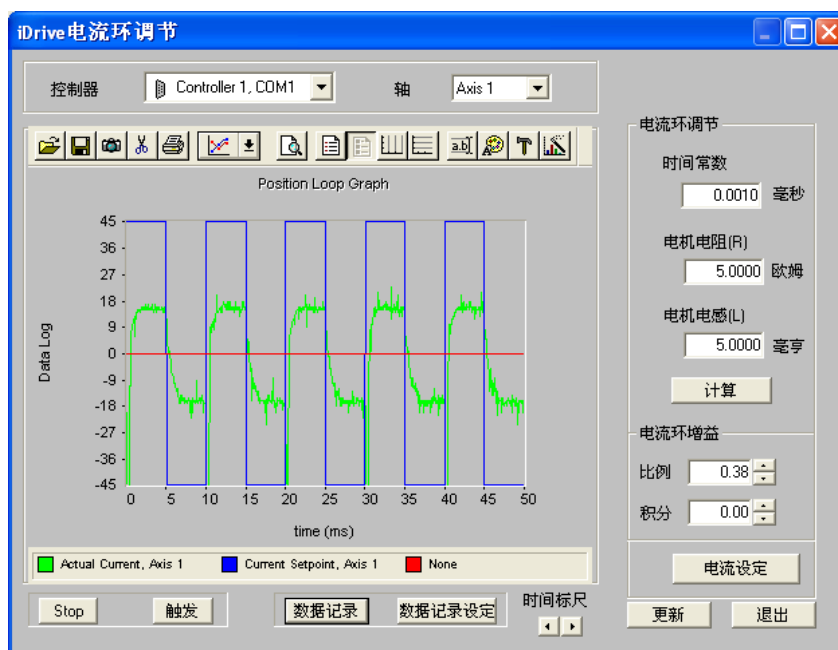


图 52. 电流环调节(1)

2) 增大比例增益而保持积分增益不变

当增加比例增益至 1.18、而保持积分增益为 0 不变时。我们可以得到如图 53 的结果，此时由于比例增益的增加而加强了电流环的控制作用，电流环的响应和误差都有了一定的改善。为了得到更好的调节结果，我们需要再次基础上调节积分增益。在调节积分增益的同时，需要视情况而定改变比例增益的大小。

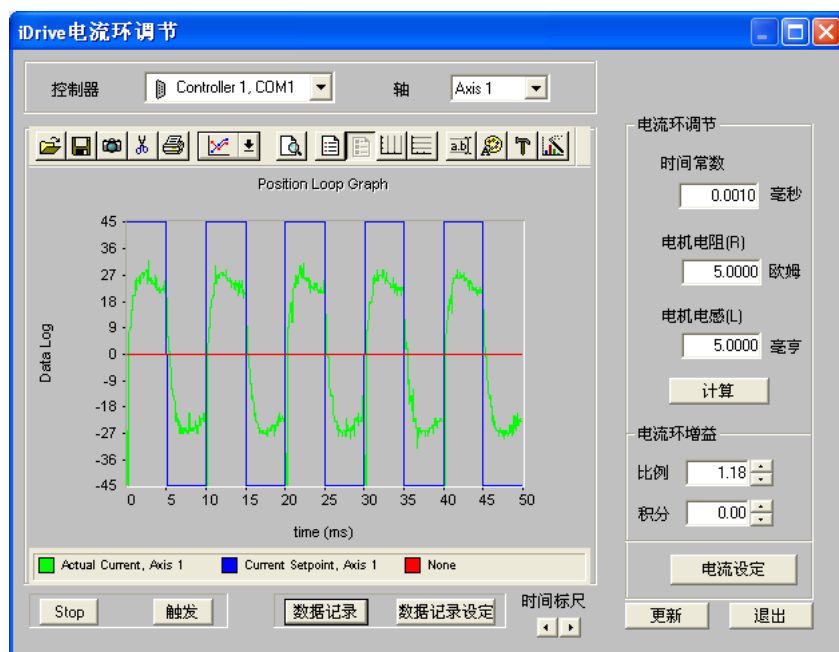


图 53. 电流环调节 (2)

3) 增大积分增益并适时增减比例增益

在上面调试结果的基础上，增加积分增益以达到减少控制误差的目的。此时将积分增益增加，而积分增益的增加除了能减少误差之外也会对系统的其它控制性能(稳定性、震荡的大小)产生一定的影响。因此，这时候需要视情况而定调节比例增益。在本次调节中，但我们把积分增益增大到 0.07 的同时，调整比例增益至 1.27，得到如图 54 的效果。

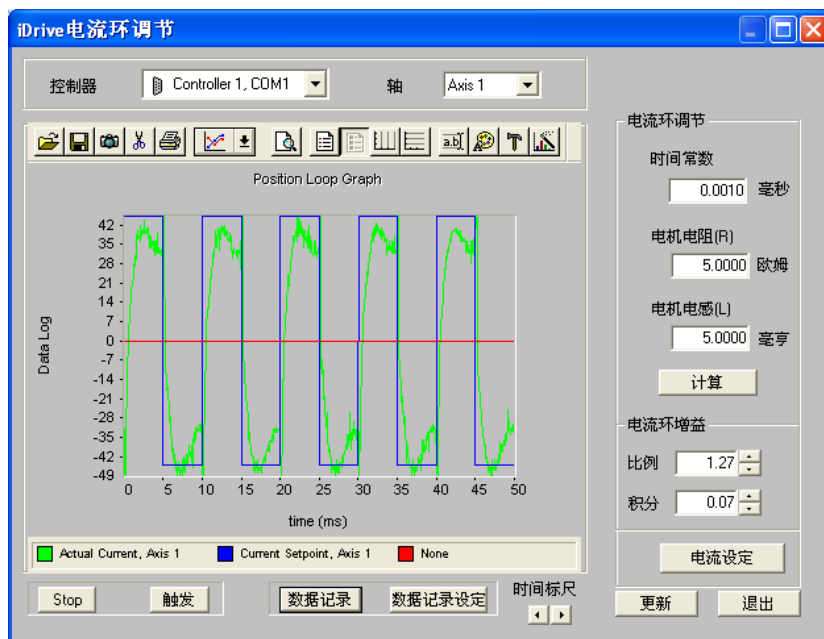


图 54. 电流环调节 (3)

4.3.5. 速度/位置环调试

电流环调节完毕之后，在此基础之上就可以进行驱动器速度/位置环的调节。速度/位置环调节主要用来完成驱动器速度/位置闭环的情况下其稳定性和控制性能的调节，包括静态性能和动态性能的调节。速度/位置环的调节界面如图 55 所示。

电流环调节完毕之后，就可以进行驱动器位置/速度环的调节。位置/速度环调节主要用来完成驱动器位置/速度闭环的情况下其稳定性和控制性能的调节。点击主界面的位置和

速度环调节对话框 ，打开如图 55 的位置/速度环调节对话框。

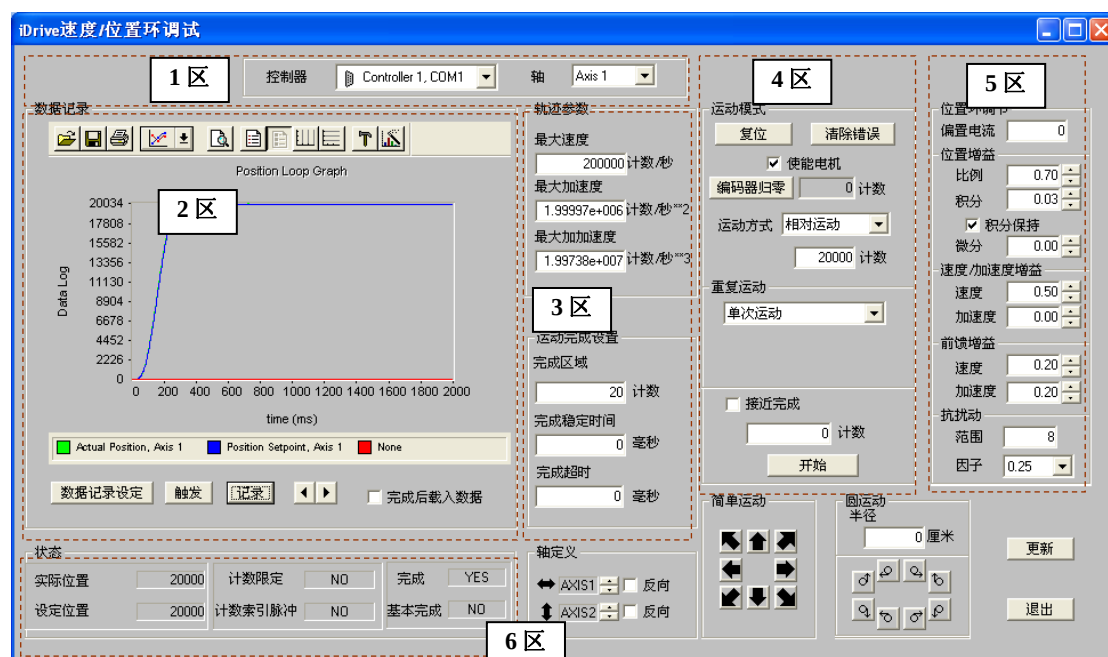


图 55. 速度/位置环调节界面

下面首先介绍一下速度/位置环调节界面的功能划分，如图 55 所示。

- 1 区：控制器和轴选择区，用于选择调试目标轴；
- 2 区：速度/位置曲线实时显示区，用于实时显示位置、速度或加速度的设定值和实际值曲线；
- 3 区：最大速度、最大加速度和最大加加速度的设置，完成区域的设置；
- 4 区：运动模式选择，用于触发和控制电机轴的运动；
- 5 区：控制增益区，用以调整控制增益的大小；
- 6 区：显示电机的当前位置以及状态。

各按钮的功能如下：

复位:	重置驱动器
清除错误:	清除驱动器故障
编码器归零:	清除编码器读数为 0
开始:	启动电机运动
触发:	触发数据记录
记录:	触发显示实时数据曲线
数据记录设定:	设置数据记录
更新:	更新参数到驱动器 Flash
退出:	退出调试界面

了解了界面的组成，我们可以开始速度/位置环的调试工作，调节步骤如下：

这里需要调节的系统参数主要有：

- 最大速度
- 最大加速度
- 最大加加速度（iDrive驱动器采用了限制加加速度功能的S曲线轨迹规划）
- 电流偏移量（一般用于垂直轴的重力补偿，默认值为0）
- 位置比例增益
- 位置积分增益
- 位置微分增益
- 速度比例增益
- 加速比例度增益
- 速度前馈增益
- 加速度前馈增益
- 抗抖动区（抗抖动区= $\pm$ （位置设定值-位置实际值），当电机实际位置进入抗抖动区以后，实际控制增益按设定控制增益乘以抗抖动因子的值进行调节）
- 抗抖动因子
- 完成区域设置（当电机运动到完成区域内时，驱动器会向上位机发送运动完成指令）
- 完成稳定时间（当电机运动到指定位置后再稳定设定的时间）

位置环的调节步骤如下：

第一步 选定增益初始值

- K_p — 位置误差增益 (如: $K_p=0.80$)
- K_i — 位置误差积分增益 (如: $K_i=0.0$)
- K_d — 位置误差微分增益 (如: $K_d=0.0$)
- B_p — 速度误差增益 (如: $B_p=0.50$)
- K_{vff} — 速度前馈增益 (如: $K_{vff}=0.00$)
- K_{aff} — 加速度前馈增益 (如: $K_{aff}=0.00$)

增益初值设定原则:

- 1). 将 K_p 和 B_p 设置到较小的值, 如0.5左右;
- 2). 将 K_i 和 K_d 设置为零;
- 3). 将速度和加速度前馈设为零;



注意: 因不同的电机会有不同的参数设定, 初值(或默认值)只能作为参考。实际的系统参数还要经过调试最终确定。

第二步 静态调节

在本手册中, 我们将电机处于锁定(使能 **Enable**)而电机没有运动的状态下调节驱动器称之为静态调节。静态调节的主要目的是调节电机处于锁定而没有运动的状态下系统的性能, 这里的性能主要有电机的震动幅度来表征。静态调节的主要目的就是在保证电机控制刚度的情况下将电机的震动调到最小或没有震动。

在这一步的调节中, 速度和加速度前馈对静态性能影响很小, 可以到下一步再进行调节。因此主要调节其它四个参数, 分别是 K_p 、 K_i 、 K_d 和 B_p 。

调节的基本原则是:

- 1). 如电机控制精度较差, 则逐步增加 K_p 和 B_p 值以增强控制作用;
- 2). 如果电机有啸叫声但是震动不是很大, 则需要减少 K_p 和 B_p 值以减弱控制作用;
- 3). 如果电机有啸叫声而且震动很大, 则需要减少 K_p 和并增加 B_p 值;
- 4). 如果系统有静态误差, 则增大 K_i 用于消除误差。但 K_i 无需设置到太大的值, 以能消除静态误差为限, 一般情况下为 $K_i=0.3$;
- 5). K_d 一般选取较小的值即可, 如 $K_d=0.01$;

电机的震动可以在6区的电机实际位置的显示中看到, 也可以用手握住电机感觉到震动的情况。

另外也可以通过调节抗抖动区和抗抖动因子来达到减少电机在静态状态下电机抖动的作用。抗抖动区和抗抖动因子的调节视实际情况而定, 抗抖动区一般选择在0到32(或更大)的范围, 而抗抖动因子则是大于0小于1的数值。

当设定的抗抖动区大小为 N 、抗抖动因子为 a 时。其作用原理如下图说示:

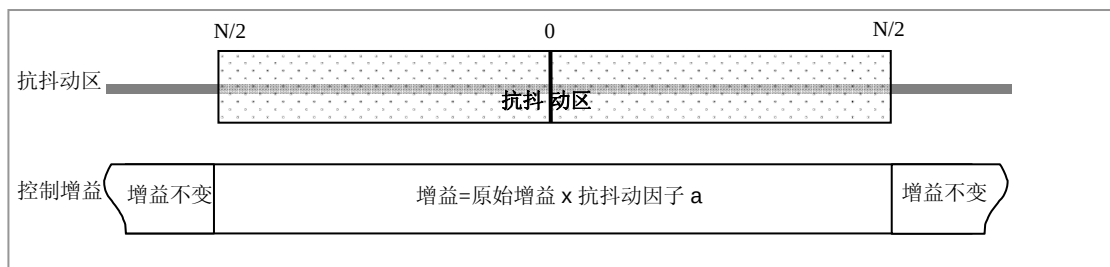


图56. 抗抖动区和抗抖动因子示例

在优化了系统的静态性能以后，我们可以来调节系统的动态性能。如同电流环调节一样，在位置环的调节中，我们也能实时看到位置环和电流环的控制效果。在调节之前首先是设定数据记录的功能。

第三步 设定数据记录

数据记录即把电机运动过程中的加速度、速度和位置的设定值和实际值记录下来。并绘成曲线，以很直观的方式显示电机运动过程中的性能。

点击速度/位置环调节对话框中的数据记录设定按钮以设定数据记录通道参数，数据记录设置对话框如图57所示。

- 1) 将目标调节轴的通道 A、通道 B 分别选为实际位置值、设定位置值；也可以通过快速设定来设定数据记录通道的选择；
- 2) 选中使能数据；
- 3) 设置采样周期为2毫秒，此采样周期大于电流环调节中的采样周期（速度/位置环的采样频率为4KHz，因此可以设置的最小采样周期为0.25毫秒）；
- 4) 设置自动触发方式为运动启动触发数据记录；
- 5) 点击更新按钮进行数据存储，然后退出数据记录设定对话框返回到位置环调节对话框。



图 57. 速度/位置环调节数据记录设定对话框

第四步 系统动态性能调节

在电机运动之前，必须进行轨迹参数设置。初始调试时，一般设定运动速度、加速度、加加速度为较小值，运动位置也设置为一较小值。调试及设定步骤如下：

1. 确保目标轴的使能电机（Motor Enable）选项被选中；
2. 点击编码器清零按钮，将初始位置设为零；
3. 将运动方式设为绝对运动或相对运动；
绝对运动：绝对坐标，以电机初始位置编码器 0 为计数起点，运动到所设定的位置值；
相对运动：相对坐标，以电机当前位置为初始位置开始运动，所运动的距离为设定的距离。
4. 设置所期望的运动终点位置值，例如 10000 编码器读数（此处编码器读数为 4 倍频以后的编码器计数）；
5. 将重复运动方式设为单次运动；
6. 设定运动的最大速度（Maximum Velocity），例如 2e+005，单位：Counts/S；
7. 设定运动的最大加速度（Maximum Acceleration），例如 2e+06，单位：Counts/S²；
8. 设定运动的最大加加速度（Maximum Jerk），例如 2e+07，单位：Counts/S³；
9. 设定完成区域，例如 50 Counts；

当电机运动到完成区域范围内（也即当真实位置与期望位置之差小于完成区域中设定值）时，驱动器将向上位机发送运动完成信息，这样上位机就可以发送下一个运动命令。

10. 完成上述设置后，点击开始按钮，电机将在上述参数下运动；
11. 当运动完成后，按钮将再次显示开始，此时可以进行新的运动。

第五步 检验运动结果

若想查看在上述参数设定下系统的运动结果，可按以下步骤操作：

点击位置环调节对话框上的数据记录按钮，将可直接观测到运动结果，如下图 58 所示。

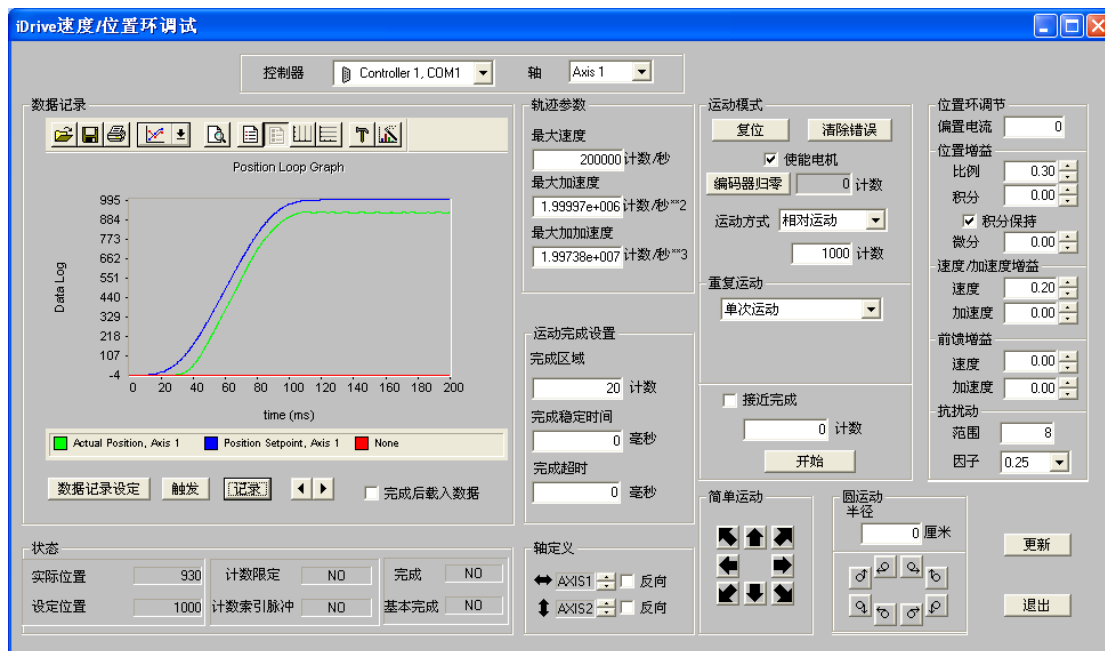


图 58. 位置环调节结果示意图（较大稳态误差）

从图 58 的结果可以看到，在上述参数下，系统的响应较慢。系统达到稳态后，真实位置与期望位置的误差较大（稳态误差较大）。因此需要继续调节参数，位置环的调节可以参考图 59 给出的流程逐步进行。

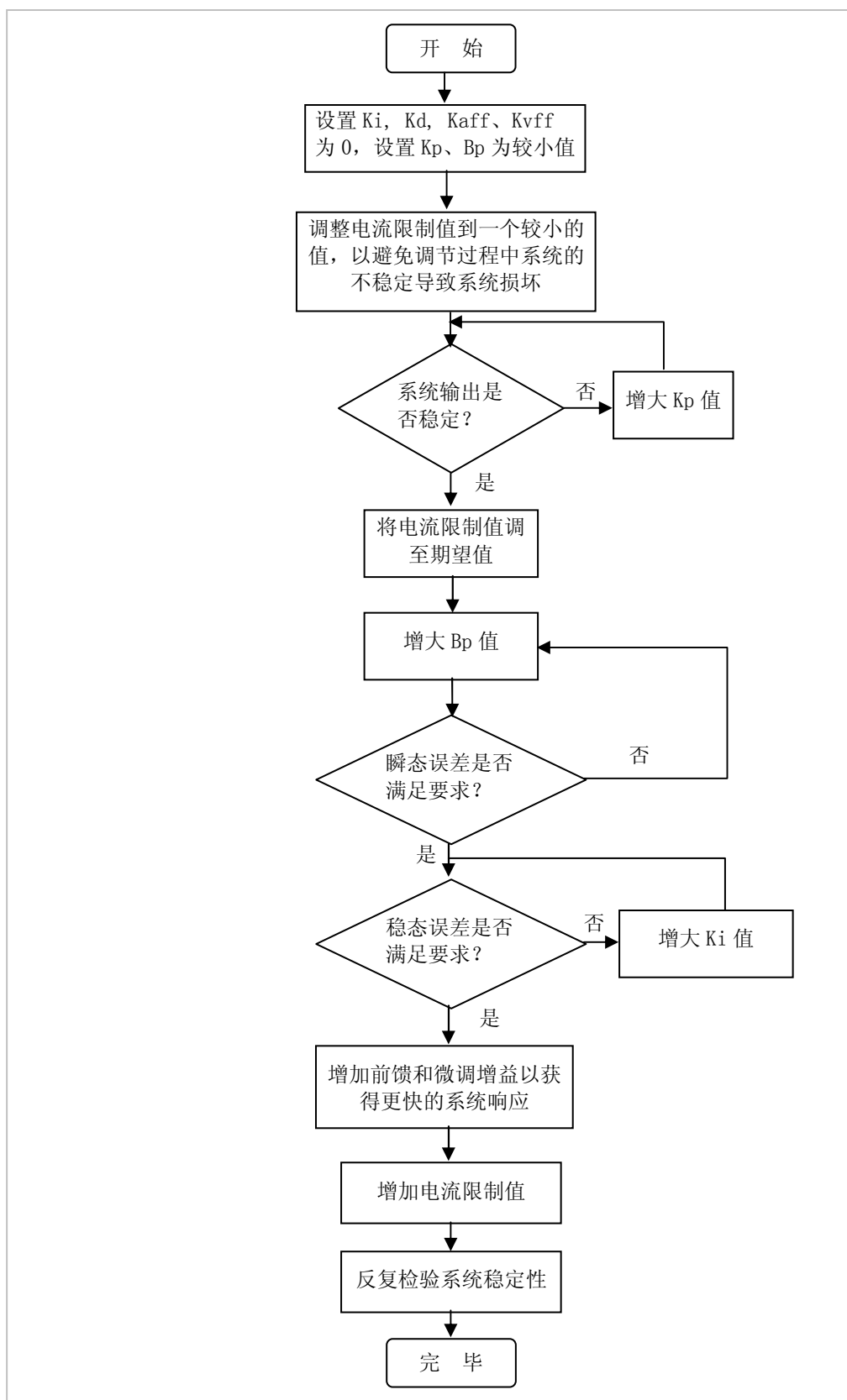


图 59. 位置环调节流程图

在前面调节的基础之上，增加比例增益 K_p 和 B_p ，系统相应的响应结果示于图 60。此时可以看到系统性能已经有了明显改善。

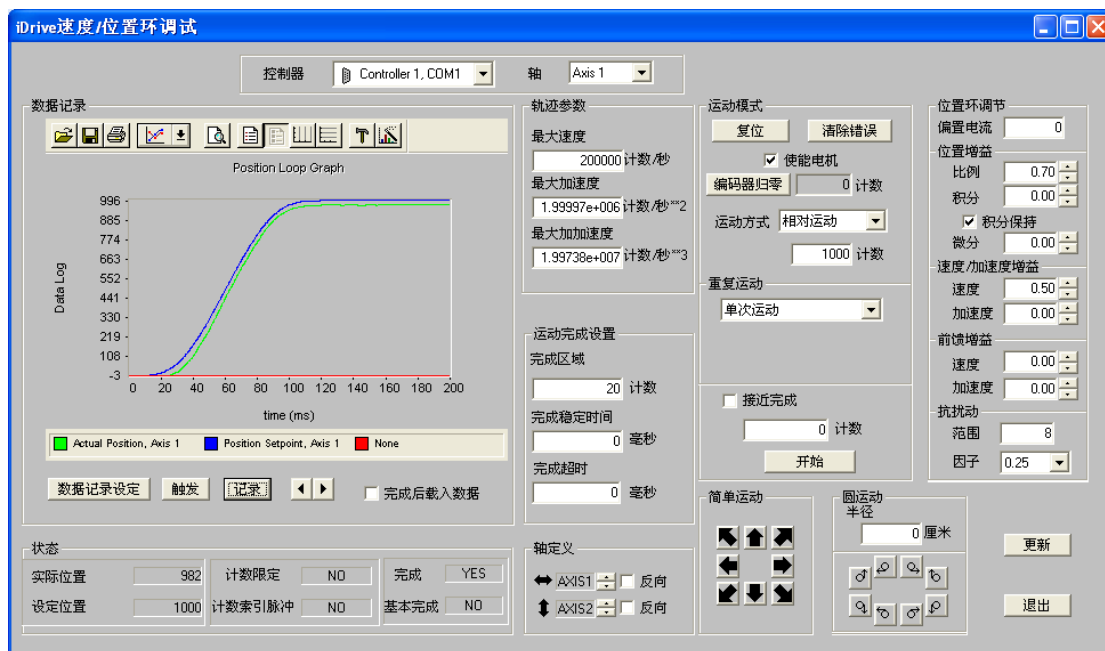


图 60. 位置环调节结果示意图（较小稳态误差）

继续增加比例增益 K_p ，并恰当调节其它控制增益。图 61 给出了一个较好的调节结果，系统响应无超调，稳态误差较小，上升时间较快。若继续调节，可能获得更好的结果。

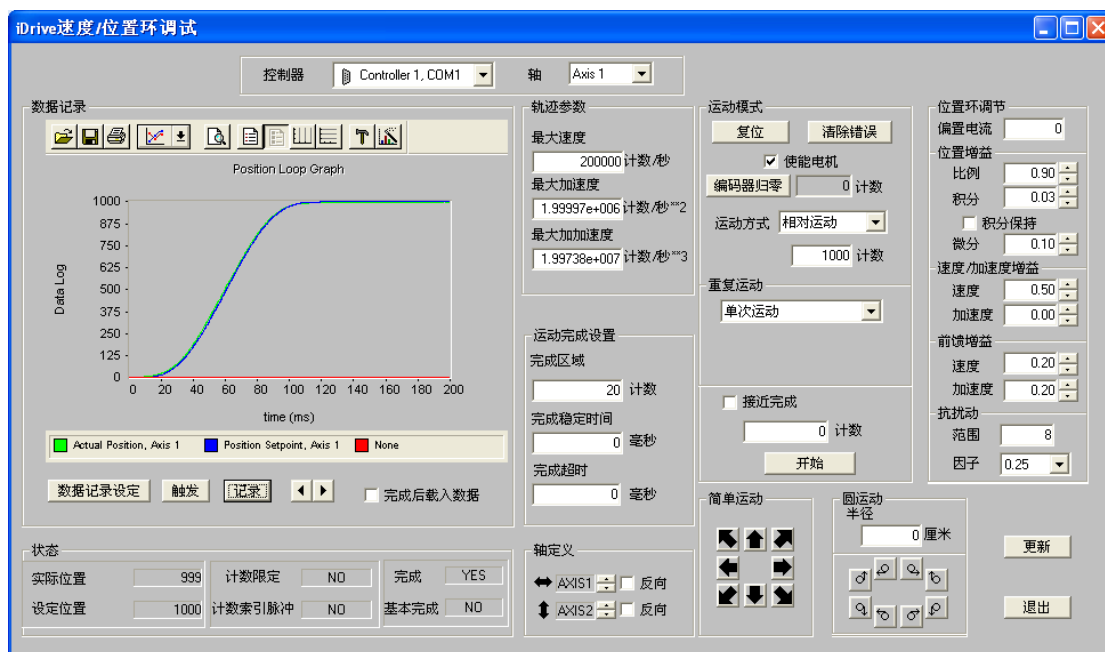


图 61. 位置环调节结果示意图（较好的调节）

图 62 给出了另一个调节结果。可以看出，该组参数使得真实位置曲线中存在较大的震荡，因此也是不满足要求的。产生震荡的原因是由于比例增益 K_p 选取过小，解决方法是适当增加 K_p 值。

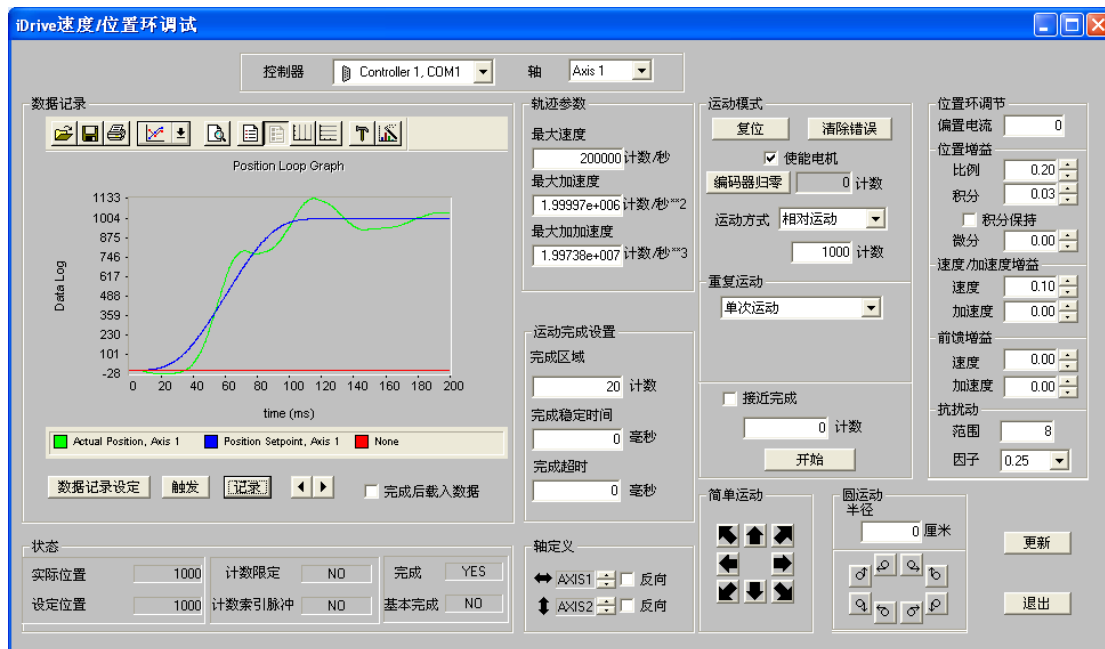


图 62. 位置环调节结果示意图（较大震荡）

iDrive 驱动器使用了控制加速度的 S 曲线运动轨迹规划，相比去控制加速度的 T 曲线而言，它有减少系统冲击的好处。在系统加速和减速的时候，速度和位置运动曲线就有更好的平滑性，不会有速度突变的情况发生。

第六步 保存所有参数

位置环调节完毕后，将 DSP 中的参数更新到 Flash ROM 中保存。

速度曲线

利用数据记录功能，我们同样可以看到电机运动过程中的速度和加速度变化的情况。下图是一个速度曲线的示例。

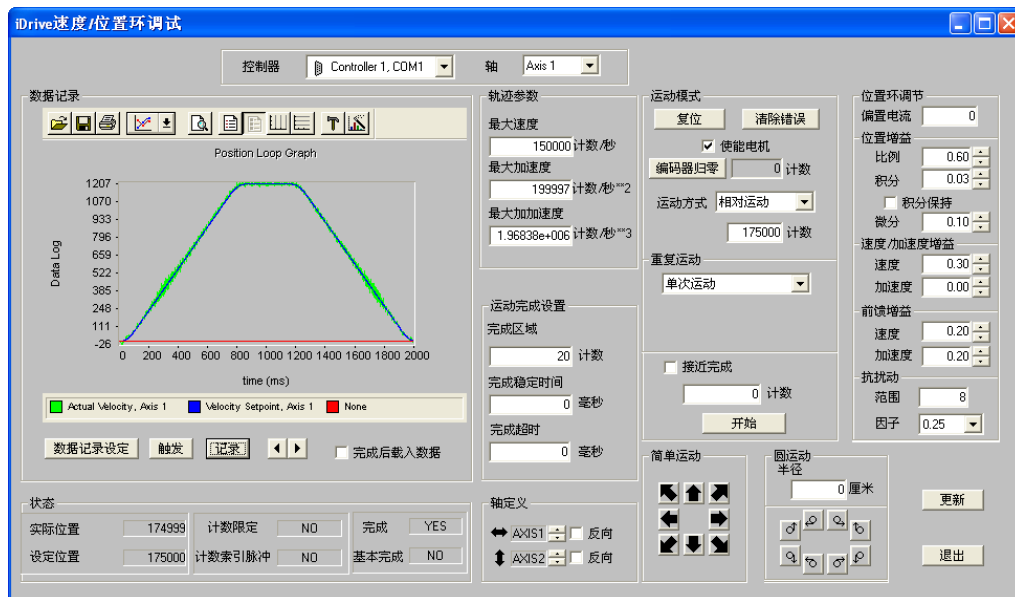


图 63. iDrive 速度曲线示例

4.3.6. 驱动器参数备份

系统调试完毕之后，我们可以将系统参数备份到计算机上。以备系统出故障的情况下用于系统恢复，参数备份文件同样可以用于调试其它同类系统之用。

系统参数备份的步骤请参考 iDrive 用户手册。

4.3.7 其它注意事项

1). 前馈控制

前馈控制有加快系统动态响应的作用，特别是对于大惯量系统加强前馈作用对系统动态性能有很好的改善作用。因此对于惯量比较大的系统，我们建议适当加强前馈控制作用。

2). 控制刚度

一般情况下，控制增益越大越能得到较好的控制刚度，系统的响应也会更好。但当控制增益较大，系统刚度比较高的时候有可能会 导致电机有震动。因此，在系统控制刚度和响应速度方面需要选择一个最佳的结合点。在保证系统控制刚度和响应速度及性能的前提下适当降低系统的控制增益，使得电机在锁定的静态状态下保持平稳无振动。

3). 电流环性能对电机控制性能的影响

较好的电流环控制性能是良好的电机控制性能的基础，电流环调节的越好，对系统控制刚度的提高无疑是有很大帮助。但是，正如 2)中所说的那样，控制增益的提高除了对驱动器的动态性能有提升之外，它对系统的静态性能(即机电处于锁定状态下的振动)也会有影响。因此在保证系统控制刚度和响应速度及性能的前提下适当降低电流环的增益，使得电机在锁定的静态状态下保持平稳无振动。

4.3.8 系统参数复制

系统参数拷贝用于两个轴之间的参数复制。当驱动器一个轴的参数设置和调试完毕以后，可以将这个轴的所有参数设置传递给另外一个没有调试的轴。极大地方便了系统调试，缩短系统调试的时间。



在系统的主界面单击参数拷贝按钮，打开如图 64 的参数复制对话框。在对话框的源轴和目标轴分别选择好相应的轴。点击复制按钮既可以开始系统参数的复制。



注意：在驱动器进行系统参数复制的时候，请不要切断电源或断开通讯。否则系统参数有可能被损坏而导致驱动器不能正常运行。后面章节所见到的系统参数更新和程序下载都有相同的情况，请注意。

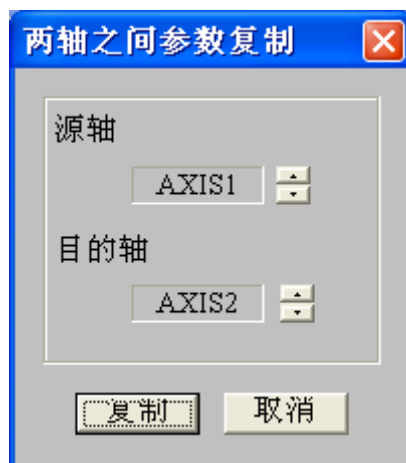


图 64. 轴间参数复制

4.3.9 更新系统参数

系统参数更新是指将驱动器 DSP RAM 中的参数保存到 Flash ROM 中，这样即使驱动器关闭电源也可以保存系统参数不被丢失。通常用于系统调试过程中或者是系统的参数已经经过修改的情况下保存参数所用。



点击主界面的更新系统参数按钮，打开如图 65 的更新系统参数对话框，选择所要操作的控制器，点击保存按钮。或者点击保存全部完成系统参数的更新。



图 65. 更新系统参数到 Flash ROM

4.3.10 保存系统参数

保存系统参数是指将驱动器中的系统参数保存到本地硬盘，生成 .MAP 参数文件。以供系统出现故障的情况下恢复系统所用。当系统在用户现场调试完毕之后，都需要作一个参数的备份，以备系统出现故障以后恢复之用。



在系统主界面点击保存参数按钮，打开如图 66 的保存系统参数对话框。选择文件所要保存到的文件夹和输入文件名，点击保存即可完成系统参数的保存。.MAP 文档的说明请参考附录。

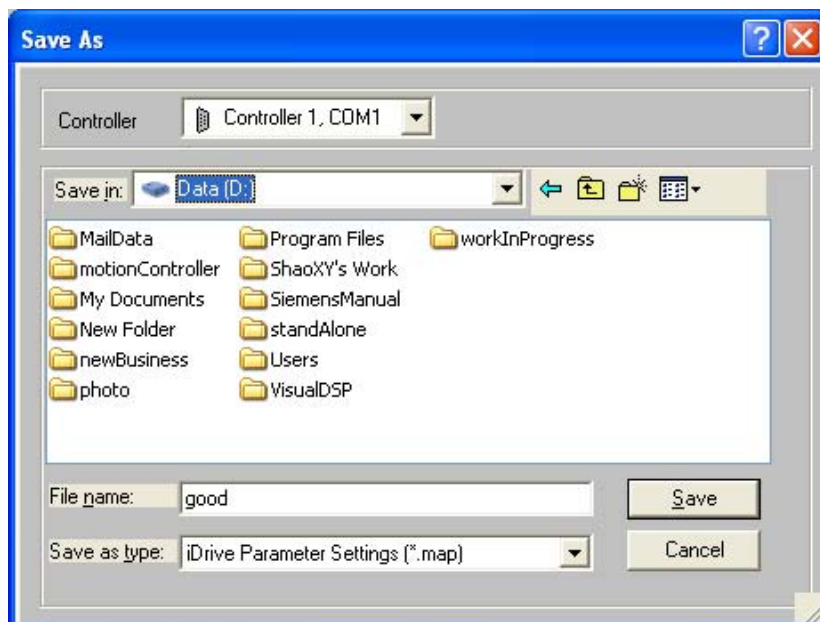


图 66. 保存系统参数到硬盘

4.3.11 故障诊断

故障诊断用于网络运行模式时判断系统的工作状态和快速故障诊断，在系统主界面点



击系统故障诊断按钮。打开如图 67 的故障诊断对话框，诊断信息显示了三主要故障类型，分别是上位机和驱动器通讯故障、驱动器故障和轴的故障。如果发现系统显示有故障，可以单击清除按钮进行故障清除。具体的故障信息请参见附录中故障诊断的相关章节。



注意：有的故障可以通过清除按钮清除故障，有的故障并不能用清除按钮达到清除故障的目的。有关故障处理具体的信息请参考附录的故障一览表。

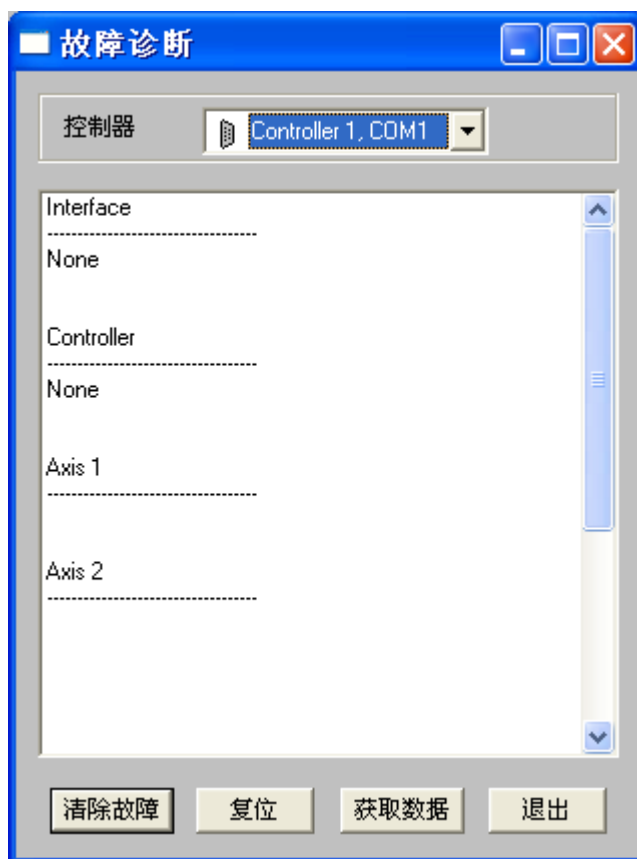


图 67. 系统故障诊断

4.3.12 下载系统参数

iDrive 的系统参数是以 .MAP 文件的形式存储于系统中，.MAP 文件所存储的内容可以参见附录的 .MAP 文件内容说明。下载系统参数就是将存储在上位机中的系统参数 .MAP 文件下载到 iDrive 驱动器中，以供系统的快速恢复或调试。



在系统主界面点击下载系统参数按钮，打开如图 68 的选择文件对话框，然后选择想要下载的 .MAP 文件，即可完成系统参数的下载。

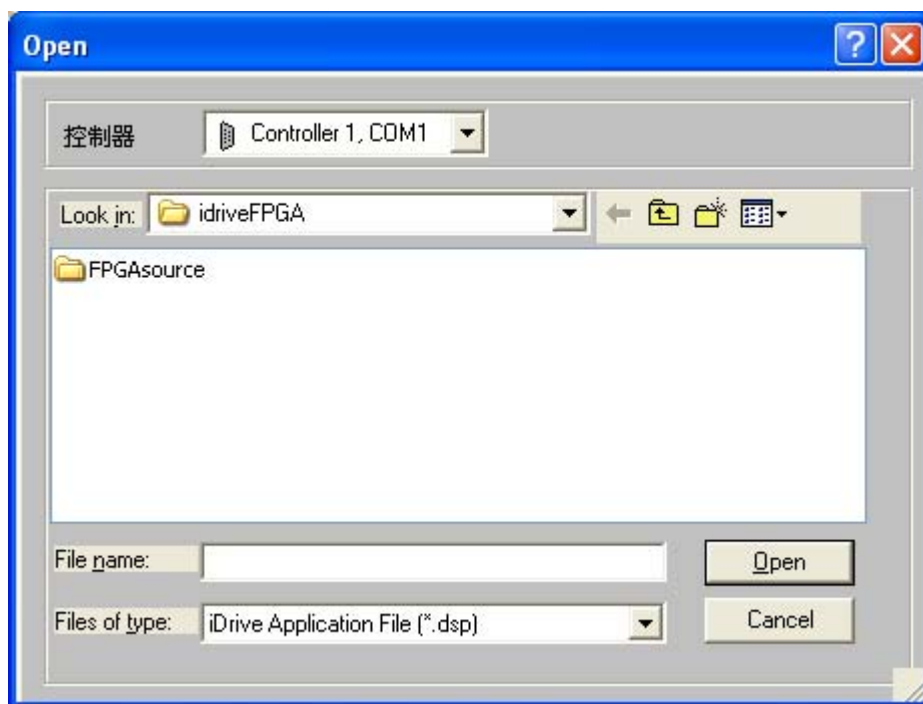


图 68. 下载系统参数 iDrive 驱动器

4.3.13 下载系统程序

程序下载功能主要用于系统软件的升级和运动程序的下载，系统软件包括 DSP 程序和 FPGA 程序。iDrive 的 DSP 和 FPGA 系统程序分别以.DSP 和.FPG 为后缀。



在系统主界面点击下载系统程序按钮，打开如图 69 的对话框。在对话框中选择所要下载的程序：

- 1) DSP 系统软件以.DSP 为后缀
- 2) FPGA 系统软件以.FPG 为后缀
- 3) 运动程序文件以.MTF 为后缀

选择需要下载的程序以后即可完成系统程序的下载，使用者无需关心程序的下载的地址。iDrive 系统会自动识别所下载的文件类型来完成下载。



注意：在下载程序的时候，请保持驱动器和上位机通讯，不要关闭驱动器的电源。如果在系统程序还没有下载完成之前关闭驱动器电源或断开网络，有可能导致驱动器无法正常工作。用户如果要更新系统程序，请先联系我们，我们的技术人员会提供最好的服务。

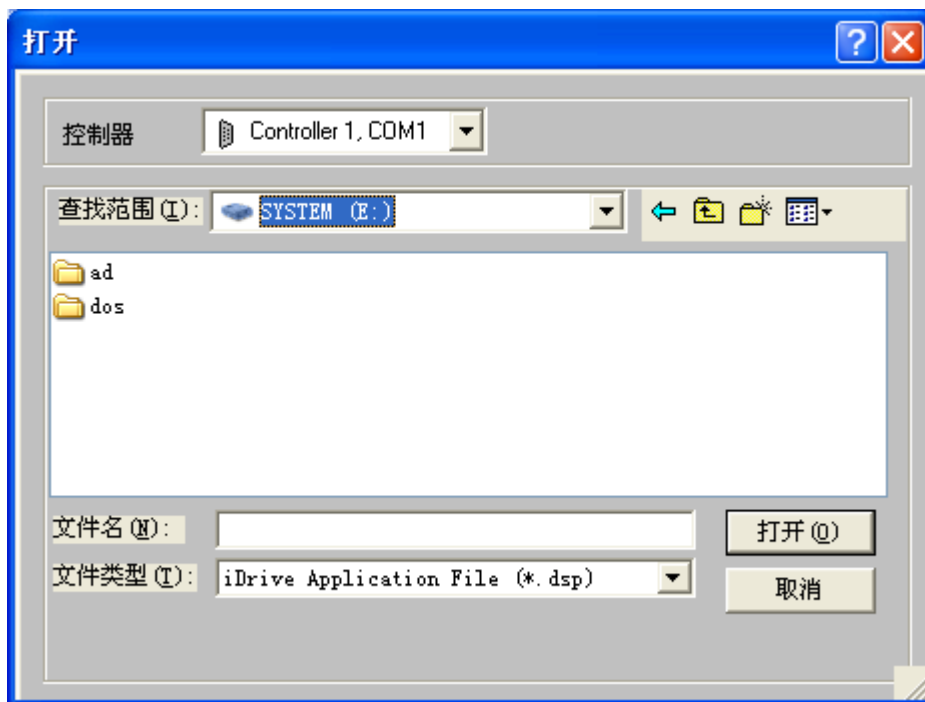


图 69. 系统程序下载

4.3.14 运动程序编程和下载

iDrive 的一个突出特点是可以脱离上位机的情况下单独运行，即 iDrive 的独立运行模式。在独立运行模式中，电机的运动方式受到存储于 iDrive 驱动器的运动程序所控制。而运动程序的触发、暂停和停止都可以由驱动器上的输入/输出口的状态来完成，这样就无需上位机的控制就能完成特定的运动控制功能，对系统成本的节约无疑带来很大的帮助。存储与驱动器内的运动程序除了用于独立运行模式之外，也可以用于网络模式下系统的调试。

在将运动程序变成界面之前，我们先来看一下独立运行模式都可以实现什么样的功能，图 70 显示了利用独立运行模式变成可以实现程序的总体框架和流程图。从流程图中可以看出，节点组成了独立运行模式运动程序的基本元素，而每个节点都有 5 个子模块，他们分别是：

[1] 和 [5]：I/O 触发控制模块；其功能是等待输入端口的状态变化或者是控制输出端口的状态变化。I/O 触发控制模块具有如下的特性：

- (1) 提供 1 个 I/O 口的条件判断或控制输出；
- (2) I/O 输入触发判断：电平或跳变沿；
- (2) I/O 输出控制：控制输出 I/O 口电平高或低；

[2] 和 [4]：延时模块即等待，单位为毫秒，最大数值为 65535 毫秒；

[3]：运动模块，即完成单轴或双轴的运动；

- (1) 单轴运动；
- (2) 双轴同时运动；
- (3) 单双轴运动可以设置不同的运动速度，但是不能设置加速度和加加速度（有上位机控制模拟是加速度和加加速度设置有效）；

图 71 和图 72 分别显示了 I/O 触发控制和输出控制的框图。

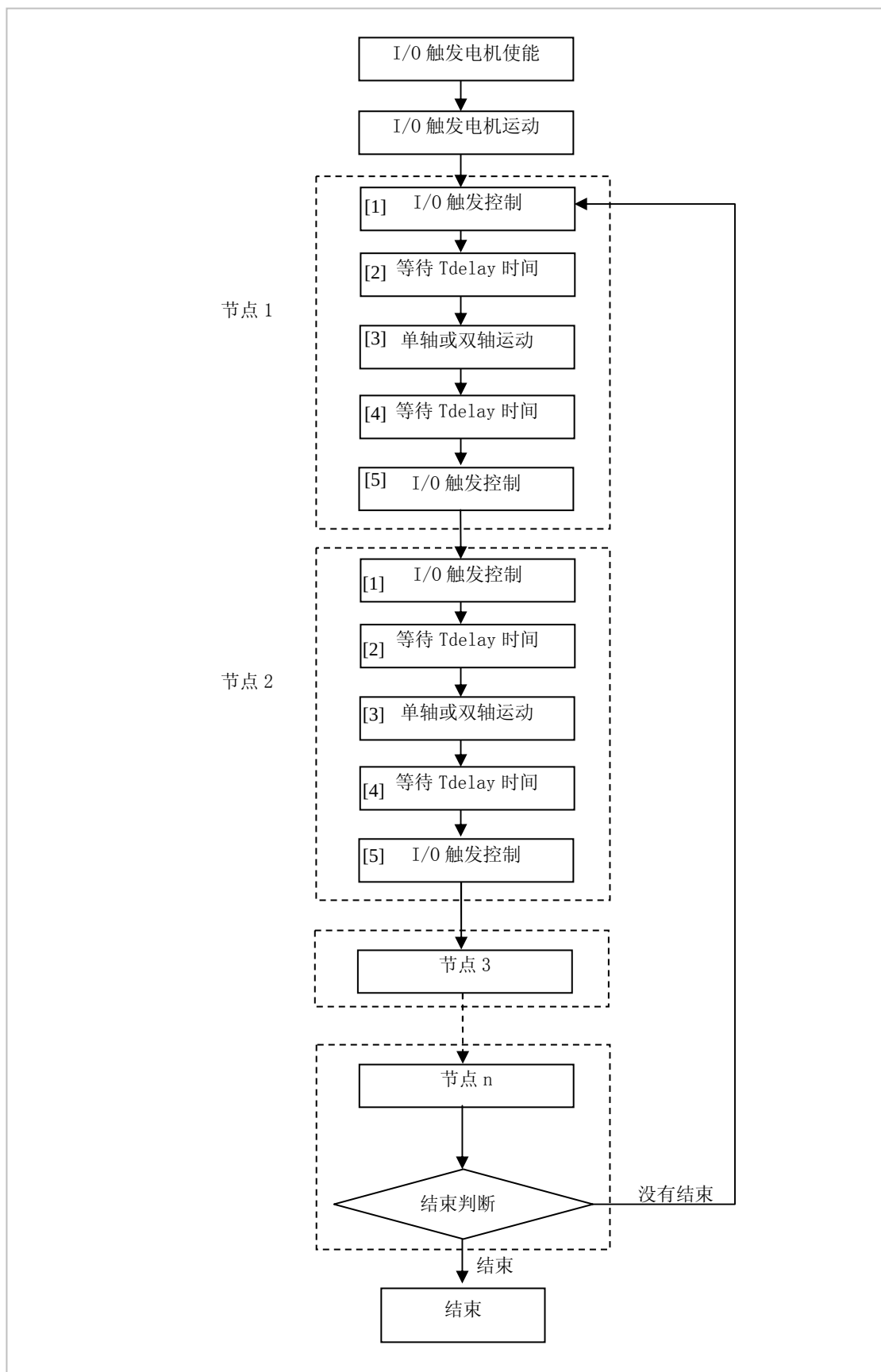


图 70. 总体流程图

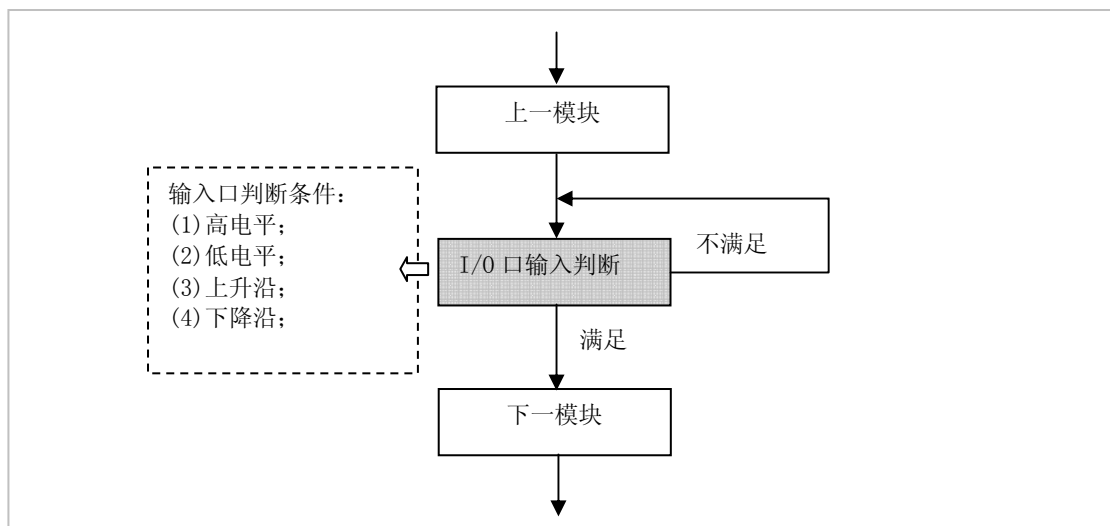


图 71. I/O 触发控制

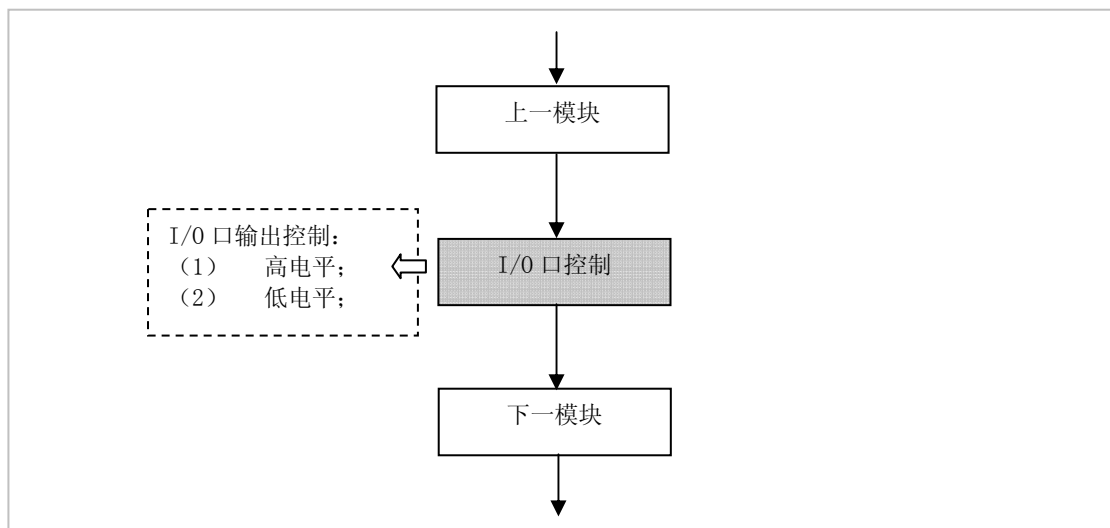


图 72. I/O 输出控制



注意：在独立运行模式下，I/O 口的 1#输入口作为电机使能之用，输入为高电平是电机使能，低电平则释放电机使能。2#输入口则作为独立运行的启动和停止信号，高电平启动运行，低电平则停止运动。



在系统主界面点击运动程序编程按钮，打开如图 73 的运动程序设计对话框。运动程序设计对话框主要由程序设计窗口、基本运动和控制信息等栏目组成。

程序设计窗口：

程序设计窗口的主要内容是显示已经设计好的运动程序序列和每个轴当前的位置信息，位置信息包括位置的设定值和当前实际值。运动程序是以图形化的方式编程和显示，



用方框表示运动程序的节点，箭头表示运动程序执行序列。在运动程序窗口下面显示当前每个轴的当前位置设定值和实际值。

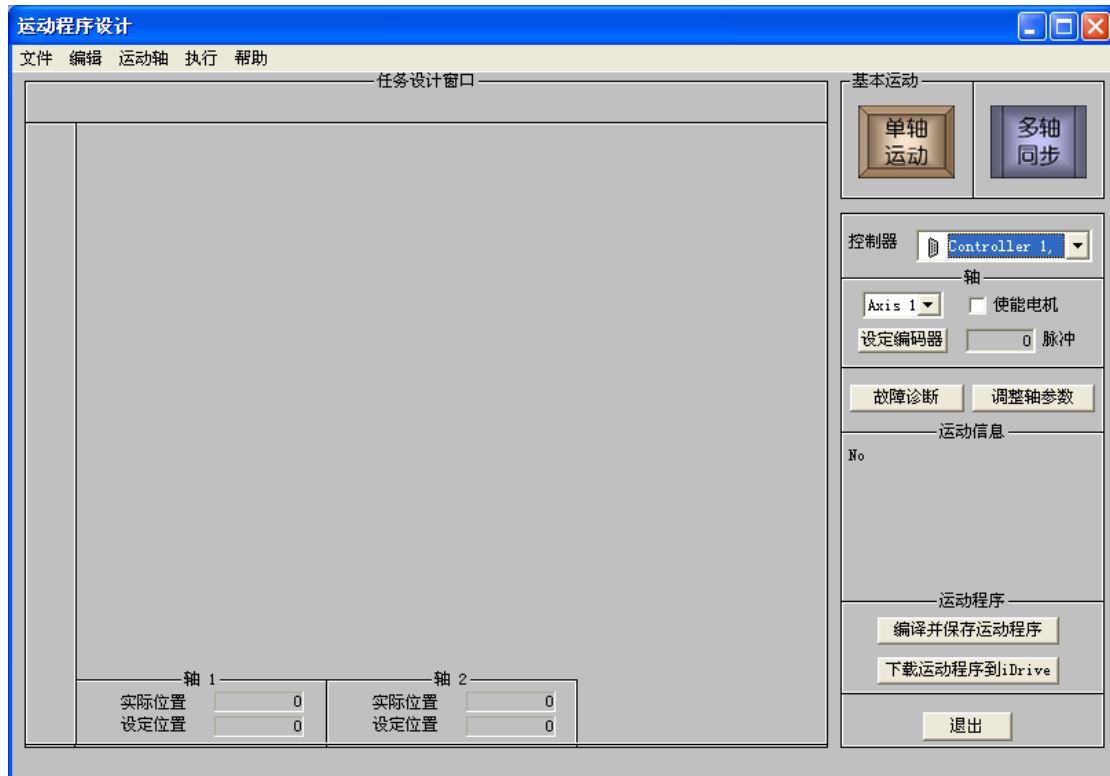


图 73. 运动程序设计界面

一个简单的运动程序如下图所示：

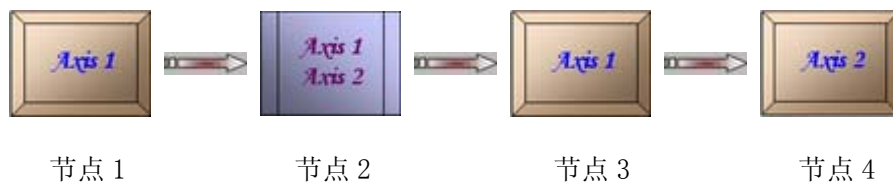


图 74. 运动程序示例

节点 1 是第一轴按设定的运动控制参数运动, 节点 2 是第一轴和第二轴按设定的参数同时运动, 节点 3 是第一轴按设定的运动控制参数运动, 节点 4 是第二轴按设定的运动控制参数运动。程序的控制按箭头的方向形成序列, 如图 74 的运动程序, 第一个节点的运动完成之后, 有第二个节点的内容控制轴的运动, 然后是第三、四个节点的运动。与每个节点的运动控制参数设定请见运动控制程序参数设定。

运动程序设计完成后可以通过选择 File->Save 菜单完成运动程序的保存，通过选择 File->Load 可以打开以前保存的运动程序。这样的运动程序除了用于独立运行模式之外，也可以在调试驱动器的时候进行两个轴的协调运动和调试。

基本运动：

基本运动主要用来进行运动程序设计时运动节点的选择，现在主要有单轴运动



和多轴运动



两个选项。点击单轴运动按钮，打开如图 75 的选择单轴运动对话框。然后选择想要添加的轴便可以实现单轴运动的添加。点击多轴运动按钮，可以打开如图 76 的选择多轴运动对话框。选择想要添加的轴，点击添加按钮便可以完成多轴运动任务的添加。

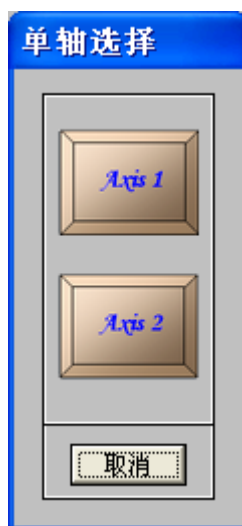


图 75. 选择单轴运动

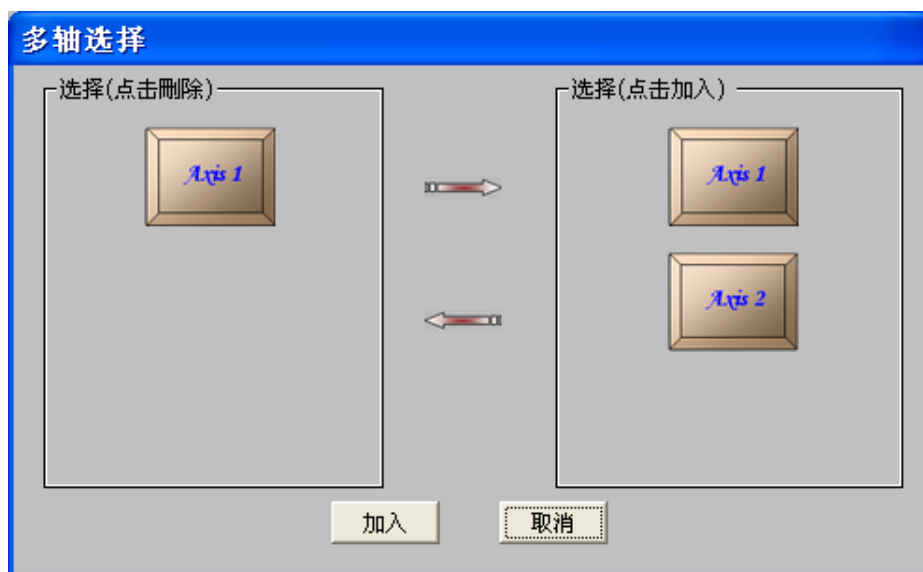


图 76. 选择双轴运动

完成运动节点的添加之后便可以进行运动参数的设定。在运动程序设计窗口中双击想要进行参数设计的节点，如果是单轴运动则会打开如图 77 的运动参数设计窗口，如果是双轴运动则打开如图 78 的参数设计窗口。在参数设计窗口中可以选择的参数有 5 部分，分别对应于每个运动节点一个步骤：

图 77. 单轴运动参数设定

- 1) 运动开始前 I/O 触发控制；
- 2) 运动前等待时间，单位为毫秒，最大数值 65535 毫秒；
- 3) 运动控制，此步骤可以设置如下运动参数：
 - 运动模式，相对运动坐标还是绝对坐标运动；
 - 电机所要运动的距离；
 - 运动最大速度、最大加速度和最大加加速度的设定（由上位机实行运动控制时，这里所提到的加速度和加加速度的设置是可以实时改变驱动器内的参数，而独立运行程序没有包括加速度和加加速度的设置值，所以只能控制不同的最大速度而不能选择不同的加加速度和加速度）；

- 4) 运动结束之后等待时间，单位为毫秒，最大数值 65535 毫秒；
- 5) 运动结束后 I/O 触发控制

参数设定完毕后点击确定便可以完成运动参数的设定。

图 78. 双轴运动参数设定

循环控制:

独立运行模式图形化编程最后一个节点是用来控制程序是否是循环执行的，如图 79 所示。此循环控制节点有四个区域的内容用来识别程序是单次执行还是循环执行。这是个区域如图 80 所示，分别是运动前时间延时、运动后时间延时、轴 1 运动距离和轴 2 运动距离。如果这是个区域的内容如图 80 所示，则程序单次执行。若如图 81 所示则是程序循环执行。



注意: 如果没有设置最后一个用于控制程序是否循环执行的节点，则程序为单次执行模式。

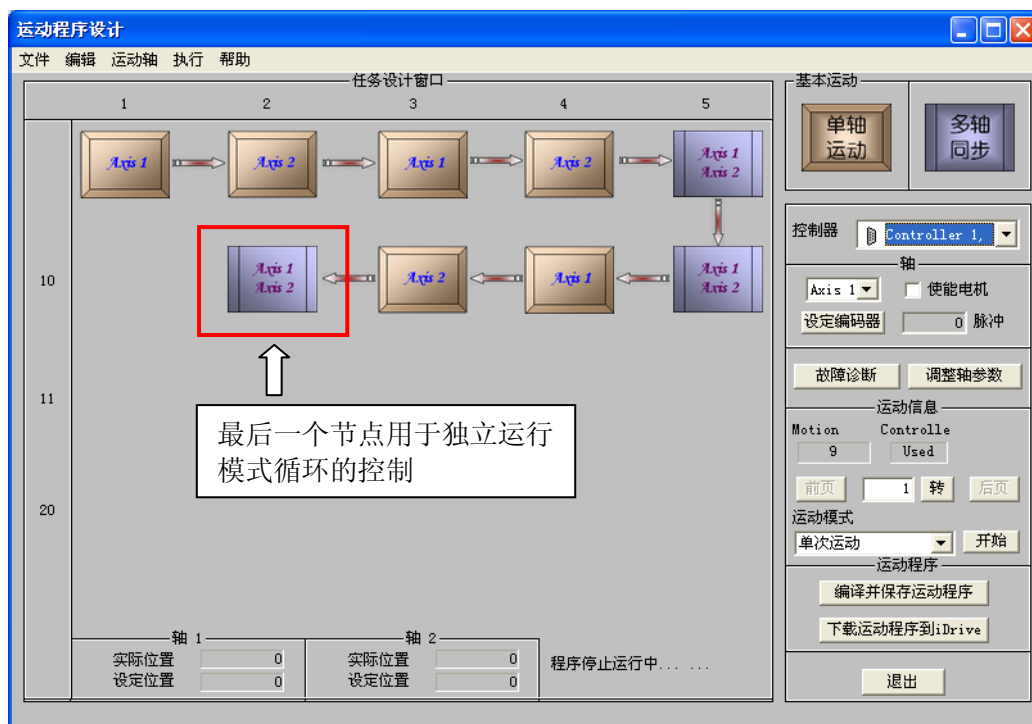


图 79. 双轴运动参数设定

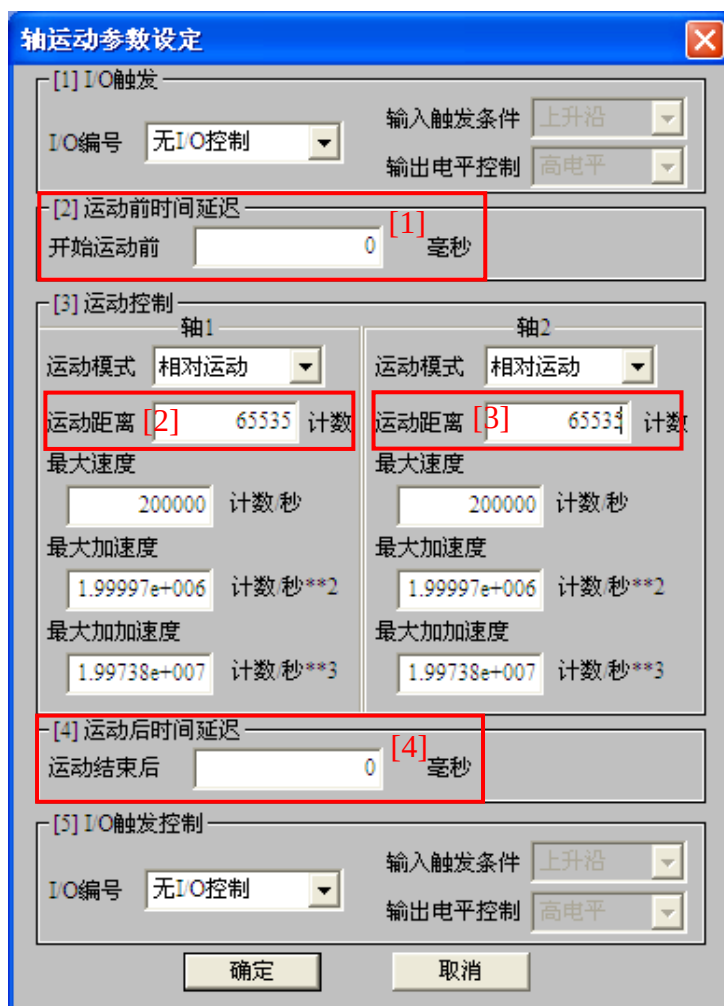


图 80. 单次执行

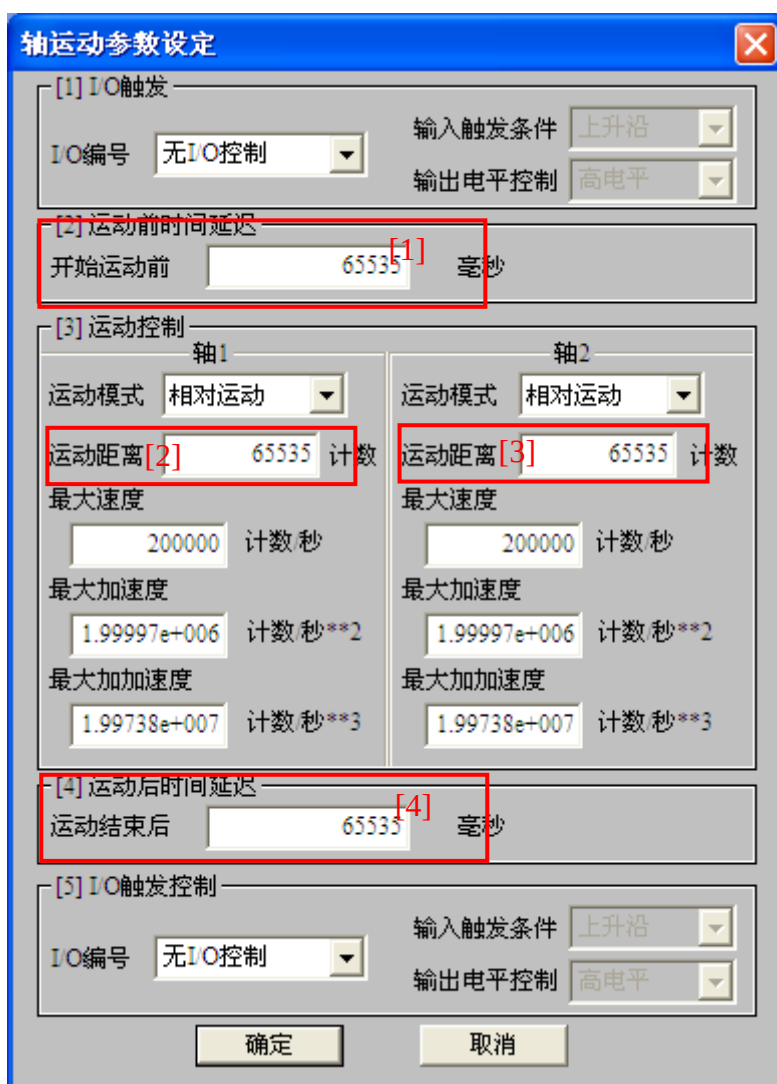


图 81. 循环执行

控制信息：

在控制信息区可以完成一些设定：

- 1) 轴的使能
- 2) 重置编码器读数为零
- 3) 故障诊断，轴参数的设定
- 4) 运动模式选择及运动的控制，包括单次运动和重复运动
- 5) 运动程序的编译
- 6) 运动程序的下载

在运动程序设计完成之后，单击运动程序生成按钮可以将设计好的图形化运动程序编译成 iDrive 驱动器可以认识的格式，运动程序文件以 .MTF 为后缀。生成运动程序以后可以点击运动程序下载按钮来完成编译后的运动程序下载到 iDrive 驱动器。

独立运行模式例程：

(1) 要实现的功能

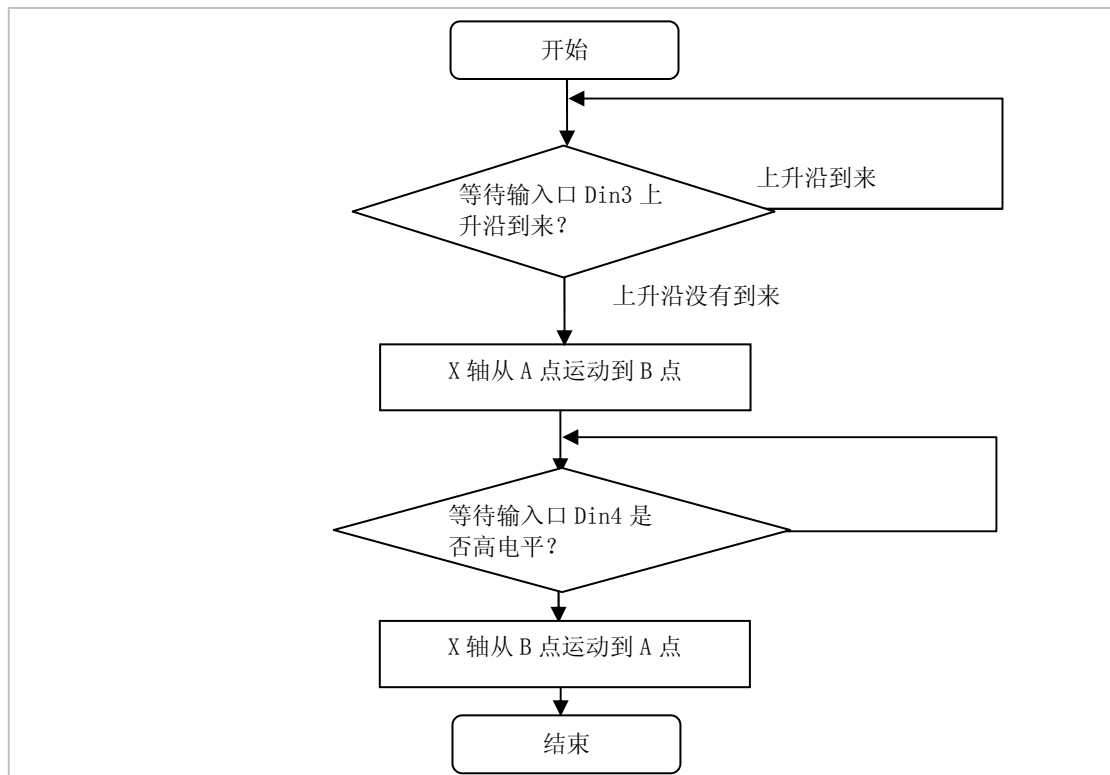


图 82. 独立运行模式运动程序例程框图

(2) 添加两个运动节点

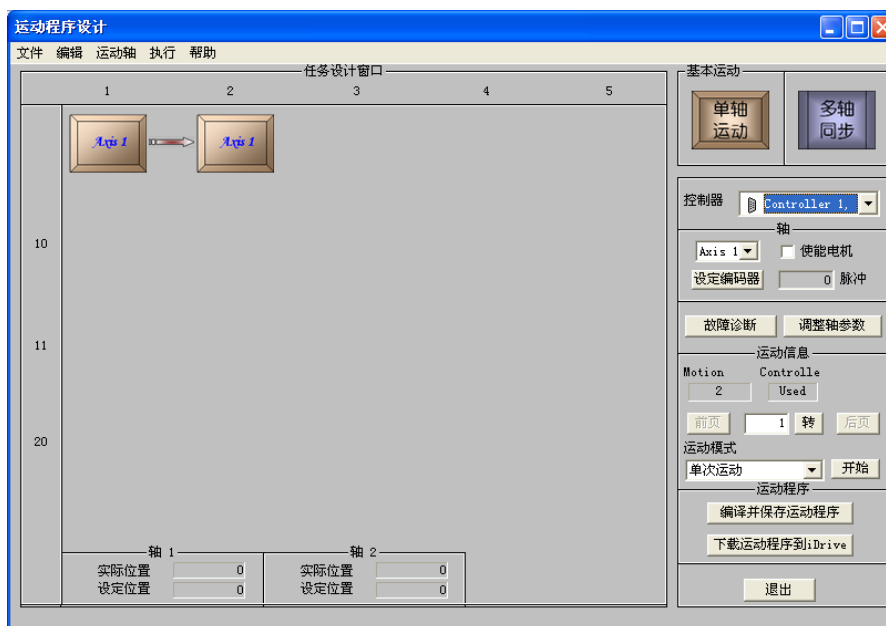


图 83. 独立运行模式运动程序节点图

(3) 设置第一个节点参数

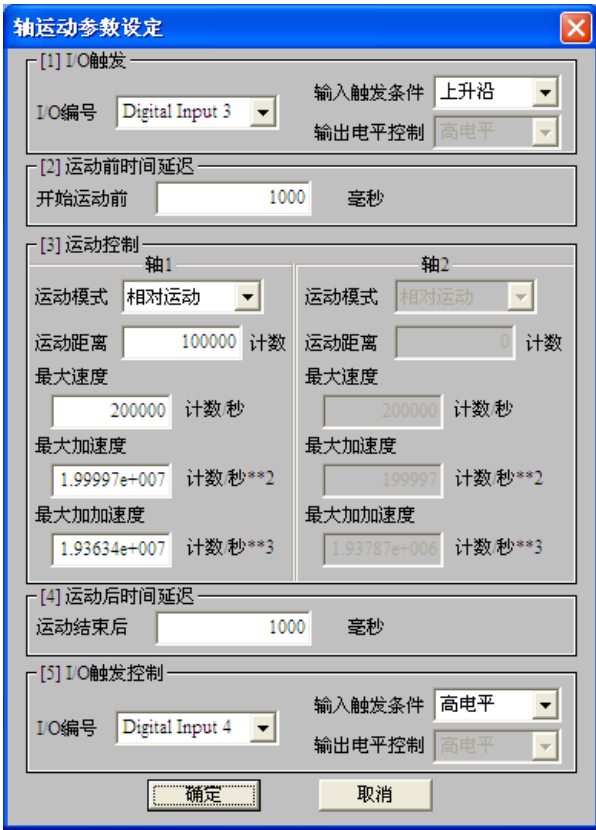


图 84. 第一个节点参数设置

(4) 设置第二个节点参数

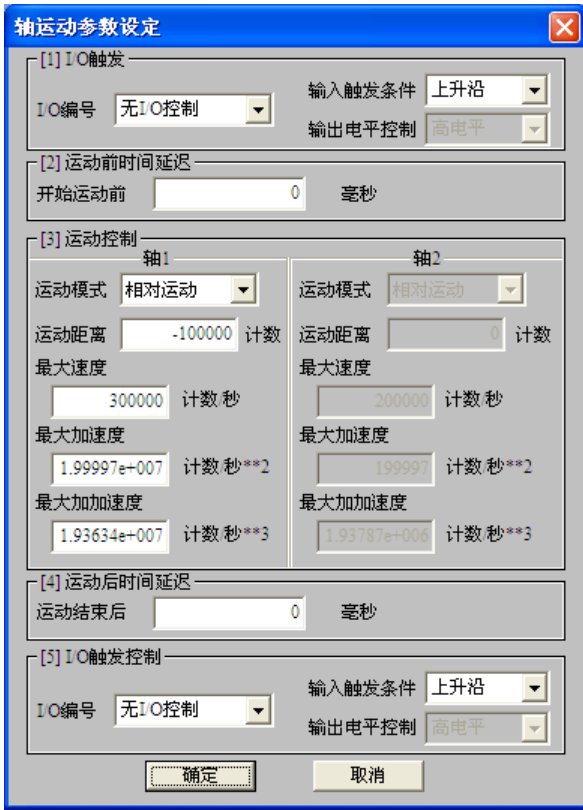


图 85. 第二个节点参数设置

(5) 编译以及程序下载运行

5. 使用 motionLib 编写用户程序

5.1 motionLib 的安装

motionLib 函数库是一个动态连接库，它主要用于网络运行模式时上位机应用程序的编程。用户可以用 VC、VB 等编程工具进行应用程序的编程。函数库分成 RS232 和 RS485 两种，RS485 动态库可以用于设置驱动器地址从而可以执行多台 iDrive 智能驱动器的联网工作。

在使用函数库进行应用程序的编程之前，需要对 motionLib 函数库进行注册安装。程序库的安装注册步骤如下：

- 1) 在和驱动器随机提供的光盘上，找到程序库 iDriveMotionLib.dll
- 2) 在操作系统桌面点击 开始->运行，打开如图 86 的对话框，输入以下命令：
regsvr32 f:\iDriveMotionLib.dll
另外，也可以打开 windos 系统的 DOS 操作界面，如图 87。输入相同的命令进行程序库的注册。
iDriveMotionLib.dll 文件的路径应根据具体情况设置

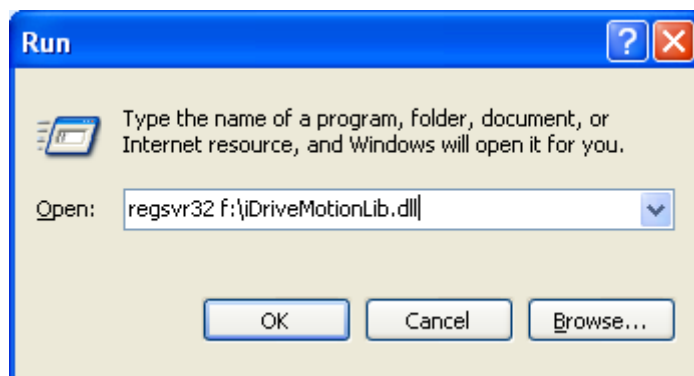


图 86. motionLib 的注册

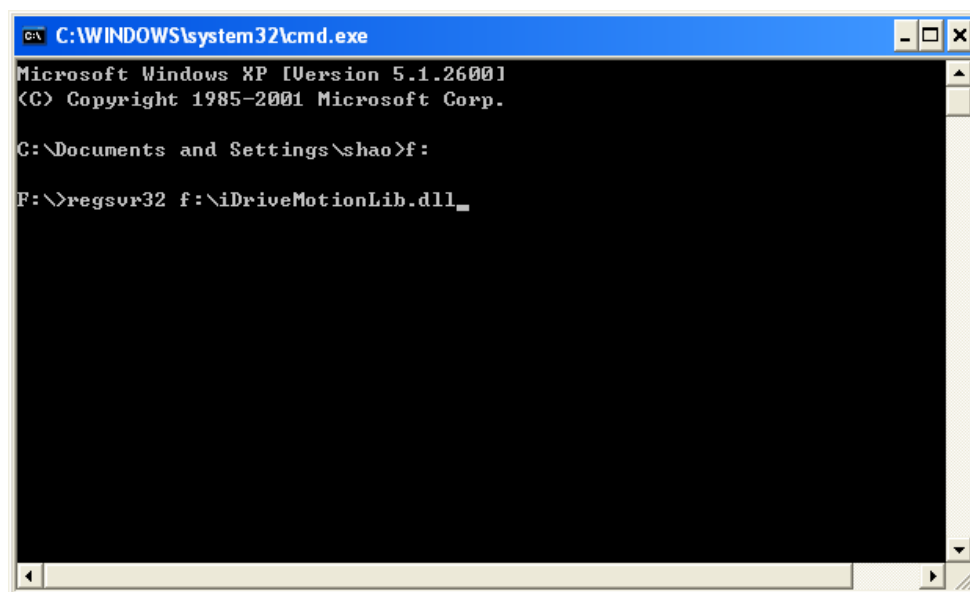


图 87. 注册 motionLib

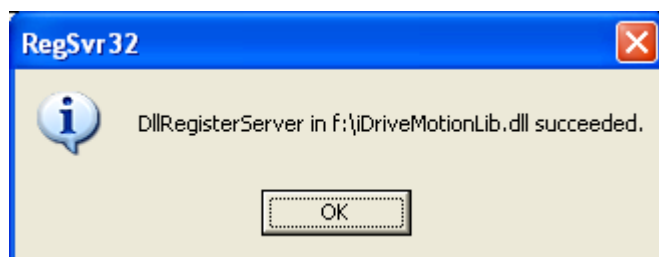


图 88. motionLib 注册成功信息

- 3) 若成功注册程序库，将弹出如图 88 的对话框。若注册失败，将出现显示其失败原因的对话框。大多数失败原因是在用户的系统内没有 msvcp70.dll 及 msucr70.dll。用户可在随机附带的光盘内找到这两个文件，把它们复制到程序库的统一路径下，然后完成程序库的注册。如果还有问题请和我们的技术人员联系。

5.2 motionLib 的结构

在使用 motionLib 进行用户应用程序的编程前,先了解一下 motionLib 的结构以及调用方法. motionLib 的结构如下图所示, 程序库一共有四个接口。这四个接口分别是 iDriveSystem, iDriveNetInterface, iDriveMotionController 和 iDriveMCAxis, 它们之间的关系如图 89, 90 所示。

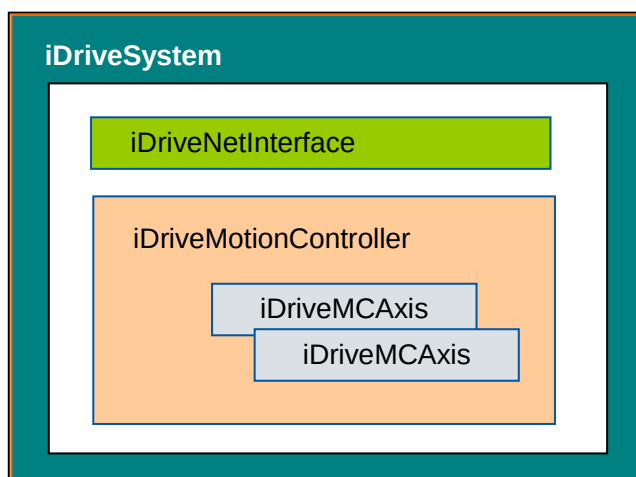


图 89. 程序库结构图

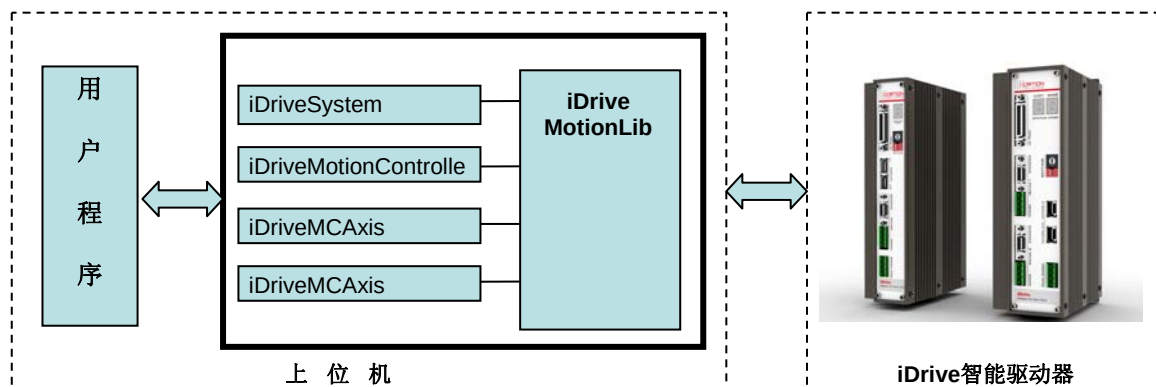


图 90. 利用 motionLib 编写用户程序并操作驱动器

各个接口的功能描述如下：

- iDriveSystem
 - 获取控制器接口
- iDriveNetInterface
 - 获取控制器网络接口
 - 通信网络相关函数
- iDriveMotionController
 - 初始化控制器
 - 控制器相关函数
- iDriveMCAxis
 - 轴初始化
 - 轴操作相关函数

在 motionLib 函数库中，系统（iDriveSystem）由两部分组成：网络接口（iDriveNetInterface）及控制器接口（iDriveMotinoController）。而控制器接口则包括两个轴（iDriveMCAxis）的控制接口。用户需要调用函数库进行应用程序编程时，须执行以下步骤：

- 1) 建立系统（iDriveSystem）；
- 2) 建立驱动器系统的网络接口（iDriveNetInterface）及 控制器接口（iDriveMontionController）；
- 3) 利用网络接口（iDriveNetInterface）进行适当的网络属性设定；
- 4) 启动（或重置）控制器接口（iDriveMontionController）；
- 5) 建立驱动器轴的接口（IiMCAxis）；
- 6) 利用控制器轴的接口对该轴电机的运动进行操作。

5.3 motionLib 函数功能说明

motionLib 提供了丰富的函数功能，为用户编写自己的上位机应用程序提供了极大的方便。下面的内容先对函数做一简单的说明，具体的函数说明请参见随机附带光盘中的函数库说明文档。

iDrive 函数库的的函数说明如下：

iDrive MotionLib 动态函数库

iDriveMotionControlSystemCreate 创建一个 iDrive 驱动器的接口

iDriveMotionControlSystem

函数名称

功能描述

get_NetInterface

获取 iDriveNetInterface 接口.

get_MotionController

获取 iDriveMotionController 接口.

iDriveNetInterface

函数名称	功能描述
get_NetworkType	检索在用的网络模式
put_NetworkType	设定网络模式
get_PortName	检索在用的网络通讯端口名称
put_PortName	设定网络通讯端口名称
get_BaudRate	检索传输速率
put_BaudRate	设定传输速率
get_Timeout	检索在用网络的超时设定值
put_Timeout	设定网络的超时值
SendCommand	输出指令到指定的控制器地址
get_Error	读取控制器的错误代码
get_ErrorDescription	读取错误代码的相关信息

iDriveMentionController

函数名称	功能描述
get_Address	检索控制器的地址.
put_Address	设定控制器的地址
Reset	重置控制器
get_SerialNumber	读取控制器序列号
get_SoftwareVersion	读取控制器的软件版本
get_SoftwareDateTime	读取软件于 DSP 编译的日期及时间
get_OptoOutput	读取指定的光电耦合输出状态
put_OptoOutput	设定指定的光电耦合输出状态
get_OptoInput	读取指定的光电耦合输入状态
get_DifferentialInput	读取指定的差分输入口状态

put_DiffOutput	设定指定的差动输出状态
get_DiffOutput	读取指定的差动输出状态
get_AnalogInput	读取指定的模拟输入口状态
get_Flag	读取指定应用标志的数值
put_Flag	设定指定应用标志的数值
get_UserFlags	读取 16 位用户标志值
put_UserFlags	设定 16 位用户标志值
get_DataLogTime	读取记录数据时间
put_DataLogTime	设定记录数据时间
get_DataLogAutoTrigger	读取记录数据自动触发功能设置
put_DataLogAutoTrigger	设定记录数据自动触发功能
get_DataLogChannel	读取记录数据通道的信息及相关轴的信息
put_DataLogChannel	设定记录数据通道的信息及相关轴的信息
TriggerDataLog	触发记录数据功能
get_DataLog	读取记录数据
SaveParameters	储存 DSP 参数到闪存
get_Axis	检索指定轴的 iDriveMCAxis 接口
KillAll	立刻禁用所有轴
ClearErrors	清除可复原的错误及用户标志
get_Error	读取控制器的错误代码
get_ErrorDescription	读取控制器的错误信息

iDriveMCAxis

函数名称	功能描述
get_Address	读取连接控制器的轴地址
put_Address	设定连接控制器的轴地址
get_AxisNumber	读取运动轴数目

put_AxisNumber	设定运动轴数目
get_Enable	检查运动轴是否已经使能
put_Enable	运动轴使能 / 取消使能
get_Parameter	读取指定位置的参数
put_Parameter	设定指定位置的参数
get_Velocity	读取在用轴的速度信息
put_Velocity	设定在用轴的速度信息
get_Acceleration	读取在用轴的加速度信息
put_Acceleration	设定在用轴的加速度信息
get_Jerk	读取在用轴的加加速度信息
put_Jerk	设定在用轴的加加速度信息
get_Brake	读取制动器状态
put_Brake	设定制动器状态
NegativeLimitEnable	使能负向位置限制
PositiveLimitEnable	使能正向位置限制
get_PositionSetpoint	读取当前位置设定值
ResetEncoderPosition	重置编码位置读数为零
get_EncoderPosition	读取指定轴当前编码器的位置
get_Halls	读取指定轴的霍尔状态信息
MoveOnePulse	输出一个脉冲，脉冲方向跟现方向相同，而且拥有稳定的速度
MoveRelative	启动电机，并使它移动到相对当前位置的距离
MoveAbsolute	启动电机，并使它移动至一特定位置
EnableNearlyDone	设定/清除接近完成标志
get_NearlyDone	检查接近完成标志是否已设定
EnableDone	设定/清除完成标志
get_Done	检查完成标志是否已设定

get_isNegativeLimit	检查负向限制标志是否已设定
get_isPositiveLimit	检查正向限制标志是否已设定
get_isIndexPulse	检查编码器索引脉冲标志是否已设定
Go	触发指定轴开始运动，并在有需要的情况下清除运动标志
Stop	电机停止运动
SeekLimit	启动电机，并使它向一特定方向运动，直至到达限位开关的位置
SeekIndex	激活电机，并使它向一特定方向运动，直至到达索引脉冲
Jog	启动电机，并使它在一特定速度运动
get_Error	读取指定轴的错误代码
get_ErrorDescription	读取指定轴的错误信息
ClearMotionFlags	清除运动标志

5.4 利用 Visual C++ 进行用户程序的编程

motionLib 函数库注册完毕之后就可以进行用户程序的编程。在下面的内容中，我们将建立一个简单的 Visual C++ 应用程序来说明如何调用函数库的函数。首先用 VC++ 建立一个新的工程，并将以下的代码导入到新建的工程中去。下列代码的功能将代码中的注释。

```
// VCTest.cpp : Defines the entry point for the console application. //

#include "stdafx.h"
#include <atlbase.h>
#include <comdef.h>
#include <stdio.h>
#include <initguid.h>

#include "..\..\Include\iMotionControlSystem.h"
// iMotionControlSystem.h 文档的路径需根据实际情况调整
////////////////////////////////////
//
// Globble variables
//定义接口
IiMotionControlSystem* sys = 0;
IiMCNetInterface* netif = 0;
IiMotionController* ctrl = 0;
IiMCAxis* axis = 0;
```

```

////////////////////////////////////
// Functions and Macros
////////////////////////////////////
// Clear up and return
// 定义故障处理
#define Fail(retCode)
do {
    if (axis)
        axis->Release();

    if (ctrl)
        ctrl->Release();

    if (netif)
        netif->Release();

    if (sys)
        sys->Release();

    return retCode;
} while (0)

//Connect to network
//定义通讯参数
HRESULT Connect(void)
{
    HRESULT r;
    //设置网络类型为 RS232
    if (FAILED(r = netif->put_NetworkType(_bstr_t("RS232"))))
        return r;
    //设置通讯超时时间
    if (FAILED(r = netif->put_Timeout(50)))
        return r;
    //设置 RS232 口
    if (FAILED(r = netif->put_PortName(_bstr_t("COM1"))))
        return r;
    //设置通讯速率
    return (r = netif->put_BaudRate(115200));
}

//主程序
int main(int argc, char* argv[])
{
    HRESULT r;
    LONG err = 0;
    CComBSTR e;

    try {
        //初始化控制器系统接口
        if (NULL == (sys = IiMotionControlSystemCreate()))
            Fail(-1);
    }

```

```
//初始化网络接口
r = sys->get_NetInterface(&netif);
if (FAILED(r) || !netif)
    Fail(-2);
//初始化控制器接口
r = sys->get_MotionController(0, &ctrl);
if (FAILED(r) || !ctrl)
    Fail(-3);

////////// IiMCNetInterface //////////
//连接控制器到上位机
if (FAILED(r = Connect()))
    Fail(-4);

////////// IiMCMotionController //////////
//重置控制器
ctrl->Reset();

//得到轴信息
if (FAILED((r = ctrl->get_Axis(1, &axis))) || !axis)
    Fail(-5);
{
    ctrl->get_Error(&err);
    ctrl->get_ErrorDescription(err, &e);
    _bstr_t ed(e, false);
    printf("0x%x\n%s\n", err, (char*)ed);
}

////////// IiMCAxis //////////
{
    LONG done = 0;

    r = axis->put_Enable(1); //使能电机
    axis->MoveRelative(10000, 1); //设定电机以当前位置为基点运动 10000 个
                                //编码器计数
                                //器计数
    axis->Go(1); //发送运动命令, 电机开始运动
    axis->EnableDone(1); //设置运动完成信号

    while(!done) //等待运动完成信号有效
        axis->get_Done(&done);
    r = axis->put_Enable(0); //使电机 disable

    axis->get_Error(&err); //查看轴的故障信息
    axis->get_ErrorDescription(err, &e); //如果有故障, 得到故障信息描述
    _bstr_t ed(e, false);
    printf("0x%x\n%s\n", err, (char*)ed); //打印相关信息
}
} catch (IErrorInfo* e) {
    _com_error err(r, e);
    _bstr_t ei = err.Description();
    printf("\n%s\n", (char*)ei);
}
```

```
}  
  
Fail(0);  
}
```

启动程序，在没有错误发生的情况下，系统将先使能电机，并且让电机运动 10000 个编码器读数，然后释放电机的使能并且查看系统的故障信息。

6. iDrive 命令一览

motionLib 是我们提供给用户编写自己的上位机应用的函数库。对于有的上位机设备不支持利用高级语言进行编程，如 PLC, HMI 等。为了支持这些设备的应用，我们同样提供了 iDrive 的命令集，以方便此类应用。关于 iDrive 运动命令的内容请联系我们的工作人员。

7. 附录

(1) 故障诊断信息

前面讲到可以用 iDrive 驱动器上的数码管或者是通过 motionStrater 进行故障诊断，驱动器的故障信息及其处理方法见下表，如果故障不能排除，请联系我们的技术人员。

附表 1. 上位机及 数码管故障诊断详情

数码管代码	故障信息	故障原因	处理方法	故障类型
8	系统没有故障			
0	系统启动未完成	系统启动过程没有完成	联系供应商	致命
1	参数检查错误	未下载电机参数或系统参数不完整	重新下载参数或联系供应商	致命
2	A/D 转换故障	2.5V 经模数转换后，输出小于 0x700 或大于 0x900	联系供应商	致命
3	霍尔信号错误	霍尔信号不正常	检查编码器连接电路接口连接情况	致命
4	跟踪超限错误	设定位置值和真实位置值之差大于用户设定的跟踪误差参数	调整跟踪误差限制或调整运动控制参数使得到更好的动态性能	错误
5	编码器故障	编码器未改变而看到霍尔变动	检查编码器连接电路接口连接情况	致命
6	速度超限错误	电机速度超出用户设定的最大速度	调整运动控制参数或将最大速度参数设定增大	错误
7	操作超时警告	在稳定时间内，真实位置没有稳定在设定的位置范围内；运动完成时间超时	调整位置稳定范围的数值及完成时间；调整运动控制参数	警告
9	驱动部分欠压	高电压过低	检查高电压是否高于 24V	错误
11	平均电流错误	平均电流和时间小于操作所需值	增大平均电流和平均电流时间	错误
A	A/D 转换超时错误	模数转换芯片不显示转换结束标志	联系供应商	致命
C	制动器短路	制动器输出电流超过 5 安培	减少电流负载	错误

E	放大器短路	A、B、C相或电机电源接头电流超过18安培	减少电流负载	错误
F	致命系统错误	主控制板软件崩溃	联系供应商	致命
H	外部电压超载 (5v)	编码器/霍尔电路保险损坏	联系供应商	致命
L	过压	高电压过高	检查是否高电压低于100V	错误
n	驱动器启动顺序错误	脉冲/方向模式下，电机在电源上升时被使能	联系供应商	致命
P	FPGA 故障	FPGA读取数据故障	联系供应商	致命
U	A/D 转换偏置故障	A/D电路对转换结束不响应或者运算放大电路存在问题	联系供应商	致命
∅	通信超时	通信电路不响应将其版本信息写入DSP中的请求	联系供应商	致命
Ⓔ	温度警告	控制器温度超过65摄氏度	关闭电源等待控制驱动器冷却，或加强制风冷装置	警告
J	温度故障	控制器温度超过75摄氏度	关闭电源等待控制驱动器冷却，或加强制风冷装置	错误

联系方式

北京诺信泰伺服科技有限公司

地址：北京市海淀区中关村南大街 9 号理工科技大厦 13 层

热线：400-600-8482

网址：www.nortiontech.com

E_mail:

sales@nortiontech.com

support@nortiontech.com